
DIPLOMARBEIT

Frau B. Sc.
Nicole Dittel

**Datenkommunikation zwischen
Elementen der industriellen
Steuerung und Aufbereitung
der Daten zur Anzeige**

Mittweida, 2011

DIPLOMARBEIT

Datenkommunikation zwischen Elementen der industriellen Steuerung und Aufbereitung der Daten zur Anzeige

Autor:

Frau B. Sc.

Nicole Dittel

Studiengang:

Elektrotechnik

Seminargruppe:

ET07wA-D

Erstprüfer:

Prof. Dr.-Ing. Dietmar Römer

Zweitprüfer:

Prof. Dr.-Ing. Swen Schmeißer

Einreichung:

Mittweida, 23.09.2011

Verteidigung/Bewertung:

Mittweida, 30.09.2011

Bibliografische Beschreibung:

Dittel, Nicole:

Datenkommunikation zwischen Elementen der industriellen Steuerung und
Aufbereitung der Daten zur Anzeige. - 2011. - 7 Seiten Verzeichnisse, 44 Seiten
Inhalt, 9 Seiten Anhänge

Mittweida, Hochschule Mittweida, Fakultät Informationstechnik und
Elektrotechnik, Diplomarbeit, 2011

Referat:

Es findet eine Kommunikation zwischen drei Simatic SPS-Stationen und einer
WinCC-Oberfläche statt. Verschiedene Werte der SPS sowie auch Zeiten
werden in einer Datenbank abgelegt. Mit InTouch werden diese Daten aus der
Datenbank wieder auslesbar sein.

Es wird auf verschiedene entstandene Konflikte eingegangen und Lösungen
beschrieben.

Es wurde sehr bedienerfreundlich programmiert und in der Arbeit ausführlich die
Vorgehensweise der Programmierung beschrieben.

Vorwort

Danksagung

Hiermit bedanke ich mich bei meinen Betreuern, Herrn Prof. Dr.-Ing. Dietmar Römer und Herrn Prof. Dr.-Ing. Swen Schmeißer für die sehr hilfreiche und fachliche Unterstützung während der Durchführung dieser Diplomarbeit.

Selbstständigkeitserklärung

Hiermit erkläre ich,
Dittel Nicole, Matrikel 12471,
dass ich diese Diplomarbeit mit dem Thema :
Datenkommunikation zwischen Elementen der industriellen Steuerung und Aufbereitung der Daten zur Anzeige
selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt wurden, sowie Zitate kenntlich gemacht habe.

_____,
Datum, Ort

Dittel, Nicole

Inhaltsverzeichnis

1 Einleitung.....	1
2 Aufgabenstellung.....	1
3 Vorbereitung.....	2
4 SPS.....	3
4.1 Hardwarekonfiguration.....	4
4.1.1 MPI-Adressierung.....	8
4.2 Bausteine.....	8
4.2.1 Organisationsbausteine.....	9
4.2.1.1 OB 1.....	9
4.2.1.2 OB 35.....	10
4.2.2 Funktionsbausteine.....	12
4.2.3 Datenbausteine.....	12
4.2.3.1 Instanz-Datenbaustein.....	13
4.2.3.2 Globaler-Datenbaustein.....	14
4.3 Zusammenfassung.....	15
5 Datenbank.....	16
5.1 Datenbank erzeugen.....	16
5.2 Datenbanktabellen.....	16
5.3 Datentypen.....	19
5.4 Tabelleninhalte.....	19
6 WinCC.....	20
6.1 Variablen.....	20
6.2 Startbild.....	21
6.3 Füllstandsregelung 1.....	22
6.4 Füllstandsregelung 2.....	23
6.5 Temperaturregelung.....	24
7 InTouch.....	25
7.1 InTouch-Projekt erstellen.....	25
7.2 Neues Fenster.....	29
7.3 Variablen erstellen.....	30
7.4 Variablen löschen.....	31
7.5 Bindeliste.....	31
7.6 WindowMaker Fenster.....	32
7.6.1 Start.....	33
7.6.2 Suchfenster.....	34
7.6.3 Regelungsfenster.....	36
8 Verbindung.....	38
8.1 ODBC-Datenquelle.....	38
8.2 Verbindungsaufbau.....	39
9 Zusammenfassung.....	40
9.1 Änderungen.....	40

9.2 InTouch.....	40
9.2.1 Runtime.....	40
9.2.2 Aktionsfenster.....	40
9.3 externer Datenbankserver.....	41
9.3.1 Datenbankbenutzer.....	41
9.3.2 Datenquelle erstellen.....	42
9.3.3 Verbindungsfehler.....	44
9.4 Ausblicke.....	44
Literaturverzeichnis.....	I
Anhang.....	A
A AWL	A
B WinCC.....	B
C InTouch-Skripte.....	E
D Quelltext erläuterung.....	H

Abbildungsverzeichnis

Abbildung 1: Kommunikation.....	1
Abbildung 2: SPS Hardware.....	2
Abbildung 3: S7-300 Aufbau.....	3
Abbildung 4: Hardwarekonfiguration der S7-300 (1).....	4
Abbildung 5: speichern und übersetzen.....	5
Abbildung 6: in Baugruppe laden.....	5
Abbildung 7: Zielgruppe auswählen.....	5
Abbildung 8: Teilnehmeradresse CPU.....	5
Abbildung 9: Teilnehmeradresse CP.....	5
Abbildung 10: Zielbaugruppen stoppen.....	5
Abbildung 11: Zielbaugruppen starten.....	6
Abbildung 12: Hardwarekonfiguration der S7-300(2).....	6
Abbildung 13: Hardwarekonfiguration der S7-300(3).....	7
Abbildung 14: MPI-Nummerierung.....	7
Abbildung 15: Hardwarevernetzung.....	8
Abbildung 16: OB 35 Befehlseingabe.....	10
Abbildung 17: OB 35 allgemein.....	11
Abbildung 18: OB 35 für Füllanlage2.....	11
Abbildung 19: Standard Library.....	12
Abbildung 20: Fehlermeldung.....	12
Abbildung 21: Datenbaustein DB41 einfügen.....	13
Abbildung 22: DB 41 als Instanz-Datenbaustein.....	13
Abbildung 23: Globaler Datenbaustein.....	14
Abbildung 24: Datenbaustein DB 1.....	14
Abbildung 25: SQL ID.....	18
Abbildung 26: SQL ID Bsp.....	18
Abbildung 27: WinCC Startbild.....	21
Abbildung 28: WinCC Füllstandsregelung 1.....	22
Abbildung 29: WinCC Füllstandsregelung 2.....	23
Abbildung 30: WinCC Temperaturbild.....	24
Abbildung 31: InTouch Projektierung.....	25
Abbildung 32: InTouch Ordernname.....	26
Abbildung 33: InTouch Projektname.....	26
Abbildung 34: InTouch Fenster.....	27
Abbildung 35: InTouch.....	28
Abbildung 36: InTouch Fenstereigenschaften.....	29
Abbildung 37: Bindeliste konfigurieren.....	31
Abbildung 38: InTouch Startbild.....	33
Abbildung 39: InTouch Aktionsfenster.....	34
Abbildung 40: InTouch Suchfensterbild.....	36
Abbildung 41: InTouch Füllanlage 1 Bild.....	37

Abbildung 42: Standarddatenbank.....	38
Abbildung 43: weitere Einstellungen.....	38
Abbildung 44: Datenquelle testen.....	39
Abbildung 45: Datenquellentest OK.....	39
Abbildung 46: Datenbankbenutzer.....	41
Abbildung 47: Neue Datenquelle.....	42
Abbildung 48: Datenquelle Authentizität.....	42
Abbildung 49: Clientkonfiguration.....	43
Abbildung 50: Standarddatenbank.....	43
Abbildung 51: weitere Einstellungen.....	43
Abbildung 52: Verbindungsfehler.....	44

Tabellenverzeichnis

Tabelle 1: Organisationsbausteine.....	9
Tabelle 2: OB 35 Einstellungen.....	10
Tabelle 3: globale Datenbausteine.....	15
Tabelle 4: Zusammenfassung SPS.....	15
Tabelle 5: SQL-Tabellen.....	17
Tabelle 6: SQL-Tabelle Zeiten.....	17
Tabelle 7: SQL Datentypen [3].....	19
Tabelle 8: Datenbank Tabelleninhalte.....	19
Tabelle 9: WinCC interne Variablen.....	21
Tabelle 10: WinCC globale Variablen.....	21
Tabelle 11: InTouchvariable.....	30
Tabelle 12: InTouch Bindelisten.....	32

Abkürzungsverzeichnis

AWL	Anweisungsliste
bzw	beziehungsweise
C	Programmiersprache
CP	Communication Processor
CPU	Communication Processor Unit
DB	globale Datenbausteine
DI/DB	lokale Datenbausteine
DW	Doppelwort
E/A	Eingabe / Ausgabe
FB	Funktionsbaustein
FC	Funktionen
HMI	Human-Machine-Interface
ID	Identifikation
InTouch	Anwenderprogramm der Firma Wonderware
jpg	Dateiendung für Bilder
MPI	Message Passing Interface
MPN	Messprotokollnummer
ms	millisekunden
MS SQL	Microsoft Structured Query Language
OB	Organisationsbaustein
ODBC	Open Database Connectivity
PC	Personal Computer
S7	Programmiersprache
SCL	Structured Control Language
SDB	Systemdatenbausteine
SPC	Statische Prozeßlenkung
SPS	speicherprogrammierbare Steuerung
STRG	Steuerunstaste
VBS	Visual Basic Script
WinCC	Windows Control Center
z. B.	zum Beispiel

1 Einleitung

Die Anforderungen in der Industriellen Steuerung werden immer komplexer. So werden Anforderungen gesetzt, bei denen ein einzelnes Gerät meist nicht mehr ausreicht.

Es werden verschiedene Software- und Hardwarekomponenten unterschiedlicher Hersteller miteinander verknüpft. Die Schwierigkeit ist die Kommunikation zwischen den bereitgestellten Programmen und den Geräten, da die Hersteller meist ihr eigenes Kommunikationssystem verwenden.

Die der Fachgruppe, Industriellen Steuerung, an der Hochschule gibt es mehrere Praktikumsplätze, bei denen verschiedene Aufgabe zu bewältigen sind.

Um den Professoren eine bessere Fehleranalyse der Praktikumsplätze zu ermöglichen, soll ein Projekt erstellt werden, welches dieses Problem gezielt lösen soll.

Die Herausforderung hierbei, lag darin, dass es mehrere SPS-Geräte und vordefinierte Softwarekomponenten gab, welche alle miteinander kombiniert werden mussten.

2 Aufgabenstellung

Es soll eine Verbindung zwischen mehreren speicherprogrammierbaren Steuerungen (SPS), einem Windows Control Center (WinCC), InTouch und einer Datenbank hergestellt werden. Zudem sollen drei verschiedene SPS angesteuert werden, sodass verschiedene Werte ausgelesen und an WinCC übergeben werden können.

Vom WinCC aus sollen Daten an die SPS übergeben werden und auch Daten eingelesen und in einer Datenbank angelegt werden.

In die Datenbank werden nur Daten aus dem WinCC gelesen und in Tabellen eingetragen.

InTouch soll Daten aus der Datenbank lesen und anzeigen können. Zudem sollen die gewählten Werte zum ausdrucken bereitgestellt werden.

Im weiteren Textverlauf wird auf die Bachelorarbeit "WinCC-Oberfläche mit Datenbankbindung und SPS-Anbindung für Medienuntersuchung im Grenzfall" eingegangen (abgekürzt wird mit Bachelorarbeit, das Kapitel und die Seitenzahl). [4]

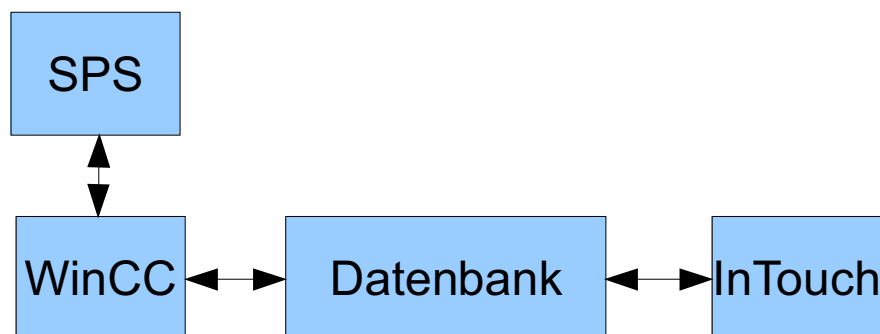


Abbildung 1: Kommunikation

3 Vorbereitung

In der Vorbereitung wurden alle benötigten Hardwarekomponenten bereitgestellt. Die verwendeten Software wurden am Rechner konfiguriert bzw. neu installiert.

Die verwendeten Komponenten sind nachfolgend aufgelistet.

Hardware:

- drei SPS S7-300
- zwei Füllstandstrecken
- eine Temperaturregelung
- Personal Computer (PC) mit der Bezeichnung AT-IS08 und dem System Microsoft Windows XP Professional Version 2002 mit Service Pack 3

verwendete Software:

- Microsoft Structured Query Language (MS SQL) - Server Management Studio (entfallen)
- SIMATIC STEP 7 Manager, Version: V5.4 + SP2
- SIMATIC WinCC, Version: V6.2.0.1 Patch 3
- InTouch der Firma Wonderware, Version: 10.1.300 1412.0130.0268.0005
- MS SQL Server 2005, Version: 9.00.2047.00

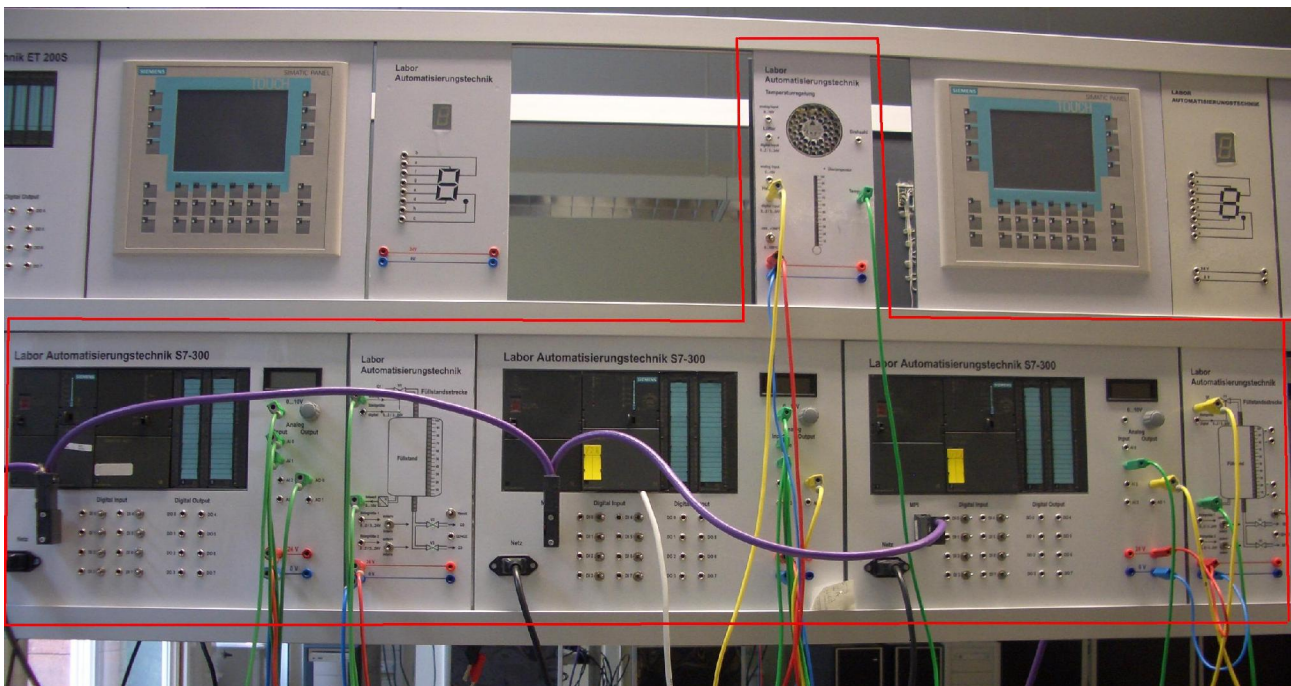


Abbildung 2: SPS Hardware

4 SPS

Eine SPS ist eine speicherprogrammierbare Steuerung. Sie wird als Steuerung in verschiedenen Anwendungsgebieten eingesetzt. So kann sie als Alarmanlage in Wohnhäusern oder zur Steuerung von Fahrstühlen und auch in der Industrie eingesetzt werden.

In Abbildung 3 sieht man den Aufbau einer, in dieser Arbeit, verwendeten S300.

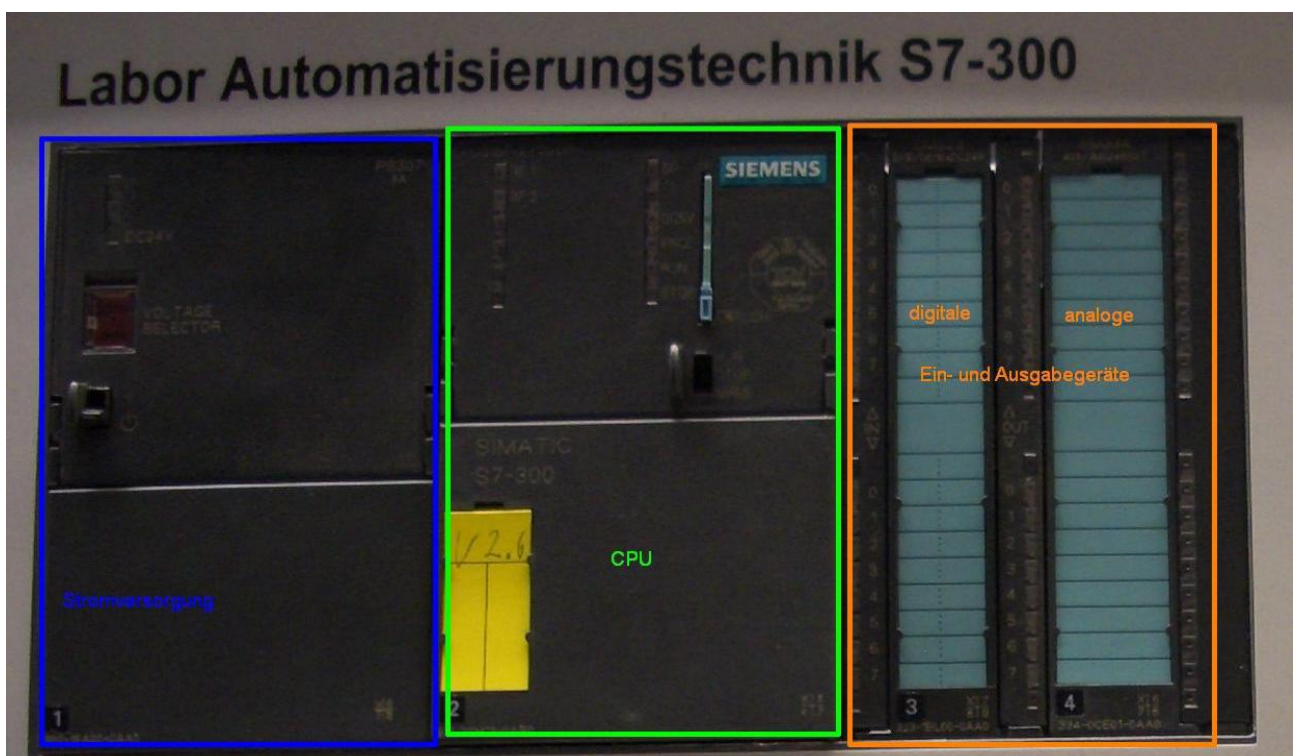


Abbildung 3: S7-300 Aufbau

In diesem Kapitel wird die genaue Vorgehensweise der Konfiguration der SPS erklärt. Man findet auch häufig den englischen Namen programmable logic controller (PLC).

Verwendet wurde die SPS S7-300 der Firma Siemens.

Zur Bearbeitung der Hardwarekonfiguration und der Programmierung der Steuersysteme wird die Software SIMATIC STEP 7 Manager verwendet.

Der Programmstart erfolgt über den Desktop Button SIMATIC Manager oder über Start \ SIMATIC \ STEP 7 \ ..., wo man zur direkten Auswahl der einzelnen Programmabschnitte gelangt.

4.1 Hardwarekonfiguration

Bei der Hardwarekonfiguration ist zu beachten, dass die SPS's einzeln konfiguriert werden müssen, das heißt es ist immer nur eine SPS eingeschaltet und bekommt ihre eigene Hardwarekonfiguration.

Um mit der Konfiguration anfangen zu können, wird der Simatic Manager gestartet und eine neue S7-300 Station eingefügt (Einfügen / Station / SIMATIC 300-Station).

Mit Doppelklick auf Hardware gelangt man nun zur eigentlichen Konfiguration.

Die Reihenfolge ist zu beachten! In der Klammer steht die genaue Bezeichnung der verwendeten Geräte (Beschriftung an den Geräten) für die erste S7-300.

Begonnen wird mit der Profilschiene (RACK 300), danach erfolgt die Hardwareeinstellung von links nach rechts (Abbildung 3). Als zweites wird die Stromversorgung eingestellt (PS-300 / PS 307 5A), weiter geht es mit der CPU (CPU-300 / CPU 315F-2PN/DP / 6ES7 315-2FH 13-0AB0 / V2.6). Bei der CPU werden noch einige Einstellung abgefragt. So ist die IP-Adresse (192.168.0.1) und die Subnetz-Maske (255.255.255.0) zu kontrollieren. zum Schluss folgen die Ein- und Ausgabegeräte. Hier wurden ein digitales Ein/Ausgabegerät (SM-300 / DI/DO-300 / SM 323 DI16/DO16x24V/0,5A) und ein analoges Ein/Ausgabegerät (SM-300 / AI/AO-300 / SM 334 AI4/AO2x8 Bit (0CE01)) verwendet.

In der Abbildung 4 wird die Hardwarekonfiguration der ersten S7-300 dargestellt.

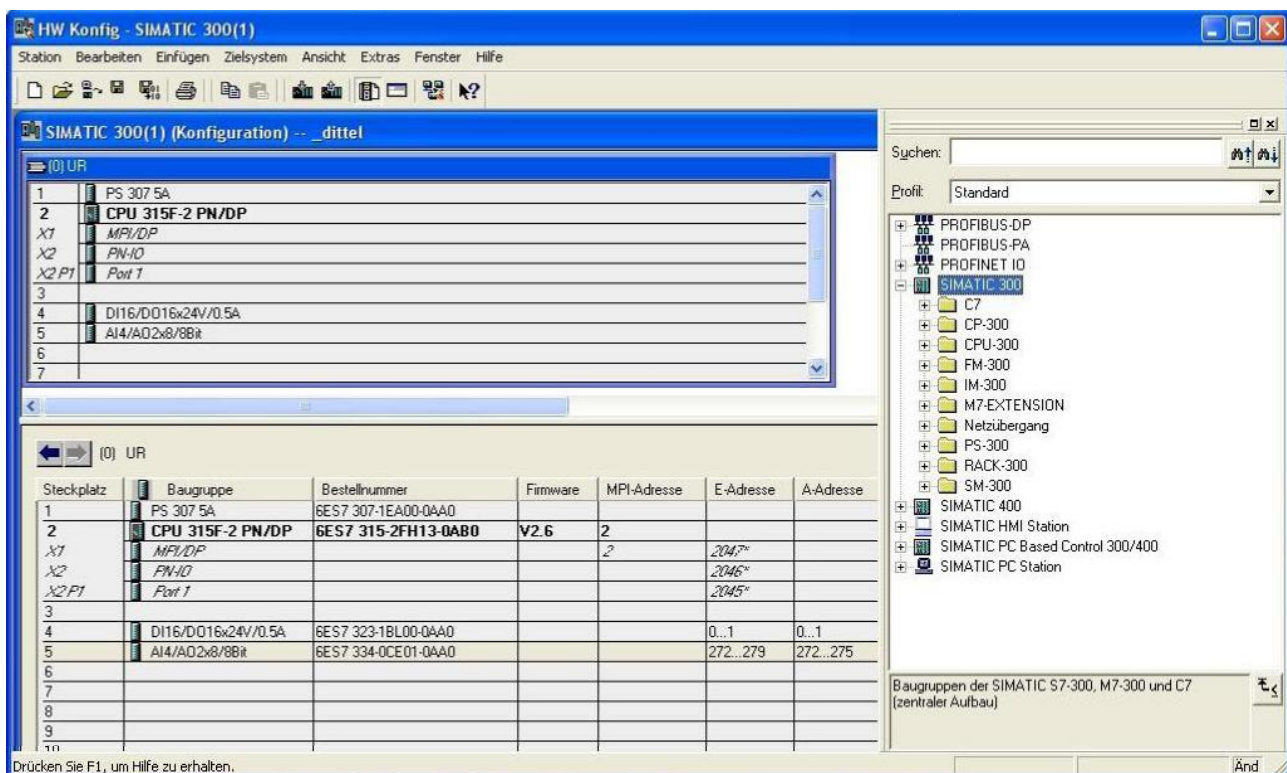


Abbildung 4: Hardwarekonfiguration der S7-300 (1)

Sind alle Geräte eingestellt, muss nun gespeichert und übersetzt werden (Abbildung 5).

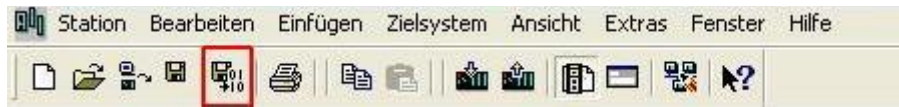


Abbildung 5: speichern und übersetzen

Nach der Übersetzung wird die Konfiguration in die Baugruppe geladen.



Abbildung 6: in Baugruppe laden

Die nachfolgenden Abbildung 7 bis Abbildung 11 zeigen das Laden in die Baugruppe. In der ersten S7-300-Station gibt es eine CPU und ein Kommunikationsprozessor (CP).

Nach der Auswahl der Teilnehmeradresse muss aktualisiert werden.



Abbildung 7: Zielgruppe auswählen

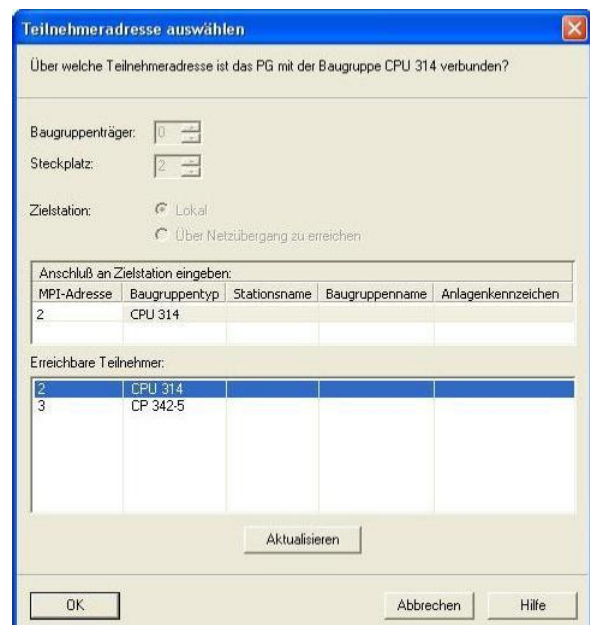


Abbildung 8: Teilnehmeradresse CPU

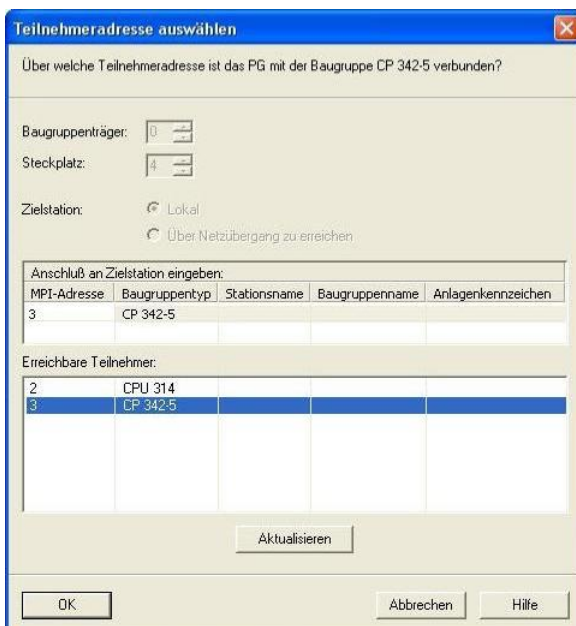


Abbildung 9: Teilnehmeradresse CP



Abbildung 10: Zielbaugruppen stoppen



Abbildung 11: Zielbaugruppen starten

Der gesamte Aufbau besteht aus drei SPS S7-300 Steuerungen. Es erfolgen somit drei Hardwarekonfigurationen, wobei immer nur eine SPS aktiv sein darf!

Gesamtbild der Hardwarekonfigurationen der anderen zwei S7-300-Station sind in den Abbildung 12 und Abbildung 13 dargestellt.

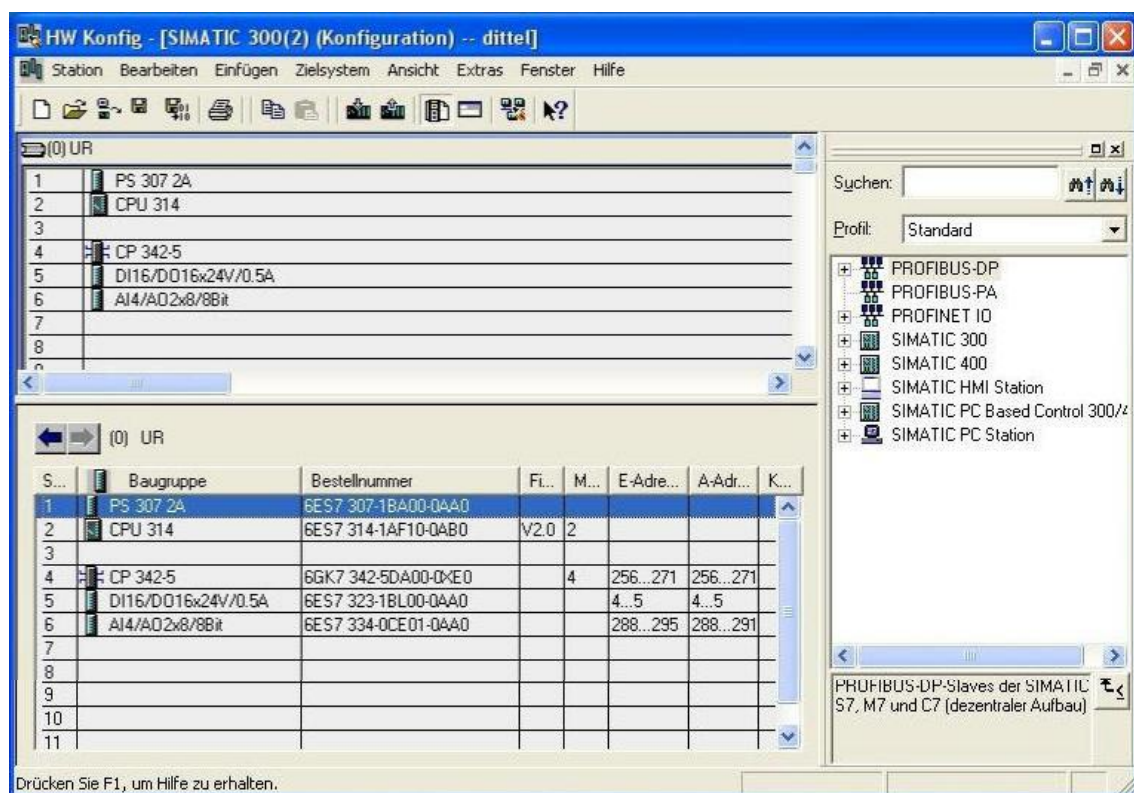


Abbildung 12: Hardwarekonfiguration der S7-300(2)

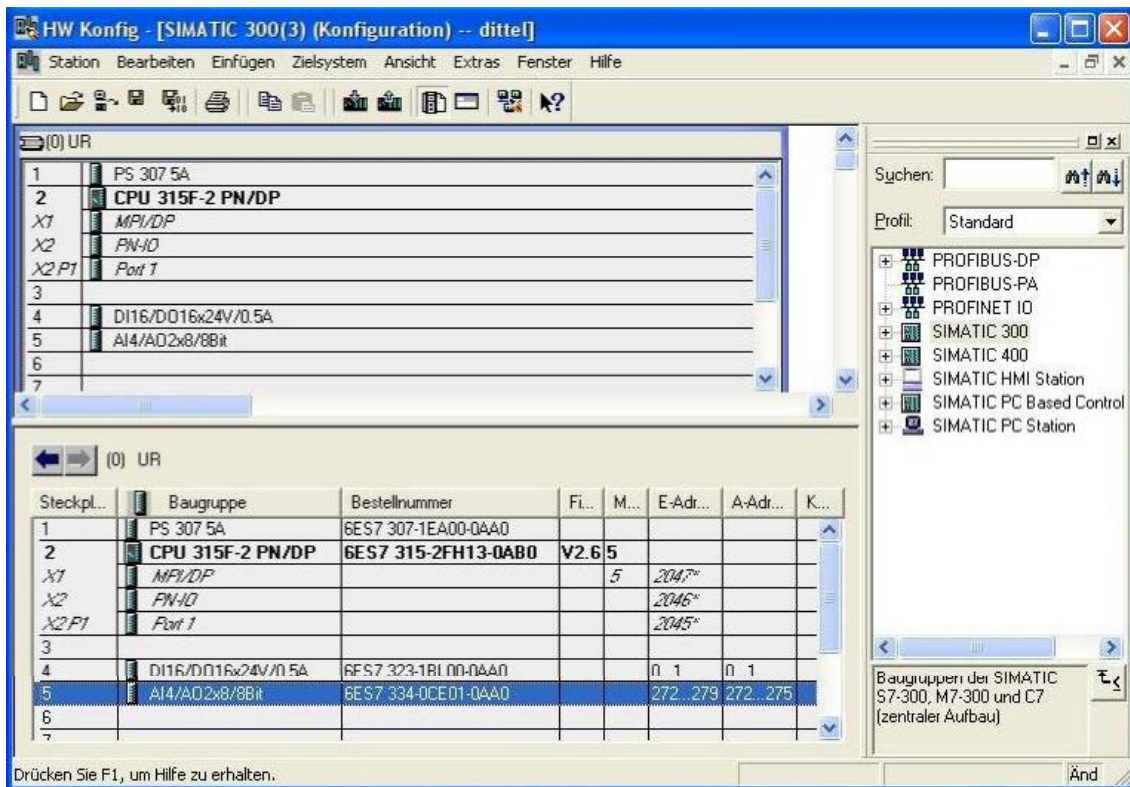


Abbildung 13: Hardwarekonfiguration der S7-300(3)

Die S7-300 Geräte sind über Message Passing Interface (MPI) miteinander verbunden. Die Zuteilung der MPI-Nummern sieht wie folgt aus (Abbildung 14).

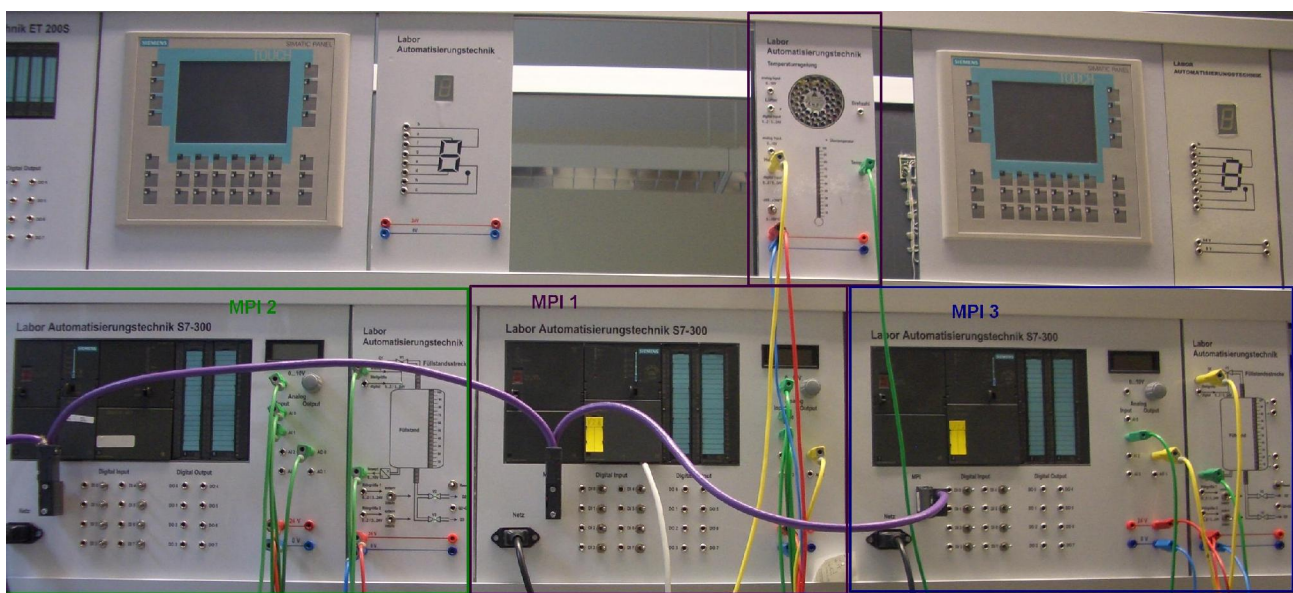


Abbildung 14: MPI-Nummerierung

Die MPI-Adressierung finden sie im nachfolgenden Abschnitt.

4.1.1 MPI-Adressierung

Schlussfolgernd sieht die Hardwarevernetzung im Step 7 Manager so aus, Abbildung 15.

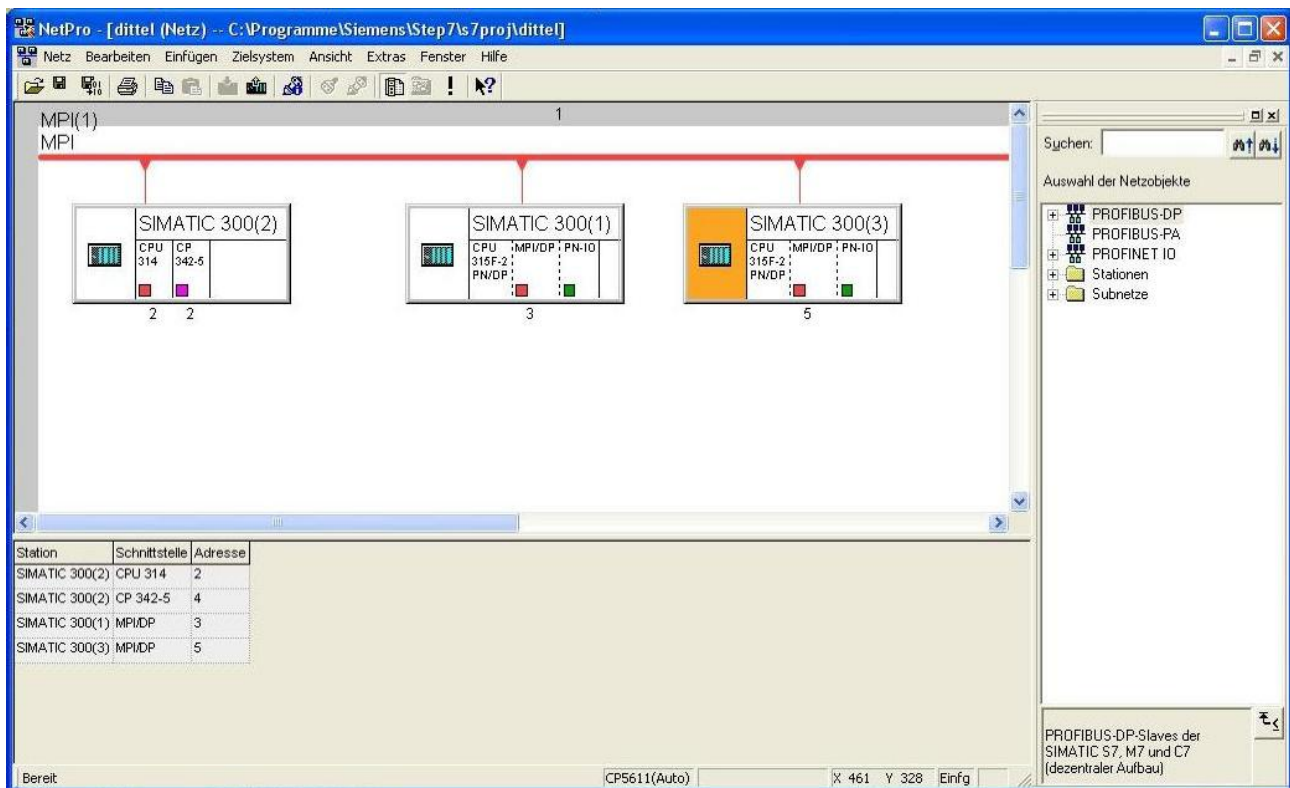


Abbildung 15: Hardwarevernetzung

4.2 Bausteine

Um eine Programmierung Durchführung zu können, werden Bausteine benötigt. Es gibt verschiedene Bausteine, u. a. Organisationsbausteine (OB's), Funktionen (FC's), Datenbausteine (DB's) und Funktionsbausteine (FB's).

Der OB ist ein zyklusgetriggelter prioritätsgesteuerter Baustein, er wird automatisch vom Betriebssystem aufgerufen. Siemens bietet vorprogrammierte OB's (Tabelle 1) [1].

Der Anwender programmiert die Bausteine in Anweisungslisten (AWL) oder Structured Control Language (SCL) oder in graphischer Programmiersprache, wie Funktionsplan, Kontaktplan, Graph oder HiGraph.

In dieser Arbeit wurde ausschließlich in AWL programmiert.

Jede SPS hat seine eigenen Bausteine, so dass es drei OB 1 gibt.

Es dürfen gleiche Namen vergeben werden, auf der selben CPU muss der Name jedoch eindeutig sein.

OB	Programmbearbeitung	Priorität
OB 1	zyklisch	1
OB 10 - OB 17	Uhrzeitalarme	2
OB 20 - OB 23	Verzögerungsalarme	3 - 6
OB 30 - OB 38	Weckalarme	7 - 15
OB 40 - OB 47	Prozessalarme	16 - 23
OB 50 - OB 51	Kommunikationsalarme	24
OB 60	Mehrprozessalarme	25
OB 80 - OB 87	Asynchrone Fehler	26
OB 100 - OB 101	Anlauf	27
OB 121 - OB 122	Synchronfehler	wie auftretender Fehler

Tabelle 1: Organisationsbausteine

Die Funktionen dienen als eine Art Unterprogramm für den OB 1. Eine Funktion liefert einen Funktionswert zurück.

Die Funktionsbausteine können Daten speichern, wenn sie einem Instanz-Datenbaustein zugeordnet sind.

Der Datenbaustein dient als Speicher, er enthält keine S7-Befehle.

4.2.1 Organisationsbausteine

Für die Regelungen und die Steuerungen, werden jeweils ein OB 1 und ein OB 35 gebraucht. Die Programmtexte können im Anhang ab Seite A nachgelesen werden.

4.2.1.1 OB 1

Der OB 1 ist der Anwenderbaustein.

Jede SPS hat seinen eigenen Organisationsbaustein OB 1.

In den Stationen 2 und 3 wird eine Füllstandsregelung realisiert. In der ersten Station findet eine Temperaturregelung statt.

4.2.1.2 OB 35

Der OB 35 wird erst programmiert, wenn die Funktionsbausteine FB 41 und der Datenbaustein DB 41 angelegt wurden. Man wählt den Menüpunkt Organisationsbaustein aus (Abbildung 21) und nennt den Baustein OB 35, zur besseren Koordinierung.

Durch die Eingabe des Befehls CALL FB 41, DB 41 (Abbildung 16) erzeugt das Programm eine Verbindung zwischen dem DB 41 und dem FB 41 und ruft diesen auf.

OB35 : "Cyclic Interrupt"

Kommentar:

Netzwerk 1: Titel:

Kommentar:

Call FB41, DB41

Abbildung 16: OB 35 Befehlseingabe

Nach der Eingabe erscheint Abbildung 17.

Folgende Eintragungen sind, je nach SPS-Station, auszuführen.

Befehl	Erläuterung	S7-300(1)	S7-300(2)	S7-300(3)
COM_RST	Neustarteingabe	E 0.0	E 0.0	E 0.0
MAN_ON	Handbetrieb	FALSE	FALSE	FALSE
PVPER_ON	Istwert wird vorgegeben	TRUE	TRUE	TRUE
SP_INT	Sollwert aus OB 1	MD20	MD20	MD60
PV_PER	Istwert (analog)	PEW274	PEW290	PEW274
LMN_PER	Stellgröße (analog)	PAW274	PAW288	PAW272

Tabelle 2: OB 35 Einstellungen

Nach den Einstellungen ergibt sich für die Füllanlage 2 das Bild 18.

OB35 : "Cyclic Interrupt"

Kommentar:

Netzwerk 1: Titel:

Kommentar:

```
CALL "CONT_C" , DB41
COM_RST :=
MAN_ON :=
PVPER_ON:=
P_SEL :=
I_SEL :=
INT_HOLD:=
I_ITL_ON:=
D_SEL :=
CYCLE :=
SP_INT :=
PV_IN :=
PV_PER :=
MAN :=
GAIN :=
TI :=
TD :=
TM_LAG :=
DEADB_W :=
LMN_HLM :=
LMN_LLM :=
PV_FAC :=
PV_OFF :=
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=
DISV :=
LMN :=
LMN_PER :=
QLMN_HLM:=
QLMN_LLM:=
LMN_P :=
LMN_I :=
LMN_D :=
PV :=
ER :=
```

```
CALL "CONT_C" , DB41
COM_RST :=EO.0
MAN_ON :=FALSE
PVPER_ON:=TRUE
P_SEL :=
I_SEL :=
INT_HOLD:=
I_ITL_ON:=
D_SEL :=
CYCLE :=
SP_INT :=MD60
PV_IN :=
PV_PER :=PEW274
MAN :=
GAIN :=
TI :=
TD :=
TM_LAG :=
DEADB_W :=
LMN_HLM :=
LMN_LLM :=
PV_FAC :=
PV_OFF :=
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=
DISV :=
LMN :=
LMN_PER :=PAW272
QLMN_HLM:=
QLMN_LLM:=
LMN_P :=
LMN_I :=
LMN_D :=
PV :=
ER :=
```

Abbildung 18: OB 35
für Füllanlage2

Abbildung 17: OB 35 allgemein

4.2.2 Funktionsbausteine

Zum Anlegen eines neuen Funktionsbausteines, wählt man den Funktionsbaustein (Abbildung 21) aus.

In dieser Arbeit wurden Funktionsbausteine aus der Bibliothek verwendet. Dazu geht man auf Datei / Öffnen / Bibliotheken / Standard Library / PID Control Blocks / Blocks / FB 41.

Abbildung 19 zeigt die Standard Library, welche ausgewählt werden muss. Die Fehlermeldung (Abbildung 20) wird mit OK bestätigt. Diesen Baustein FB 41 kopiert man in sein Verzeichnis.

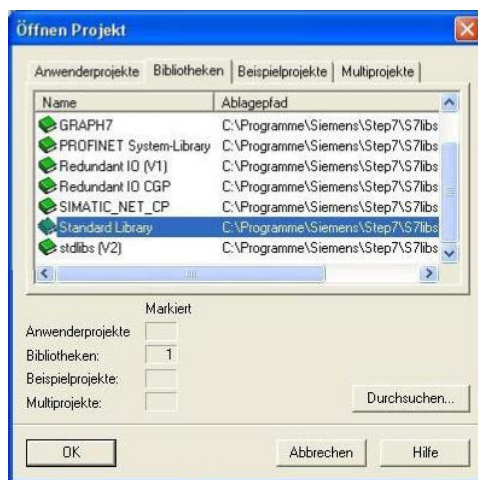


Abbildung 19: Standard Library



Abbildung 20: Fehlermeldung

Datei / Schließen beendet die Bibliothek

4.2.3 Datenbausteine

Es gibt drei Typen von Datenbausteinen im Step 7:

- Systemdatenbausteine (SDB)
- globale Datenbausteine (DB)
- und lokale Datenbausteine (DI/DB) (Instanzen)

Systemdatenbausteine sind direkt vom System vorgegeben.

Globale Datenbausteine werden für die Kommunikation mit anderen Soft- bzw. Hardwaren verwendet.

Ein Instanz-Datenbaustein wird immer mit einem Funktionsbaustein erzeugt. Es werden nur lokale Variablen verwendet [2].

Ein Austausch eines globalen Datenbaustein und eines Instanz-Datenbaustein erfolgt über: TDB Tauschen DB ↔ DI

4.2.3.1 Instanz-Datenbaustein

Nun wird ein Datenbaustein als Instanz-Datenbaustein zum FB 41 erzeugt (Abbildung 21 und Abbildung 22).

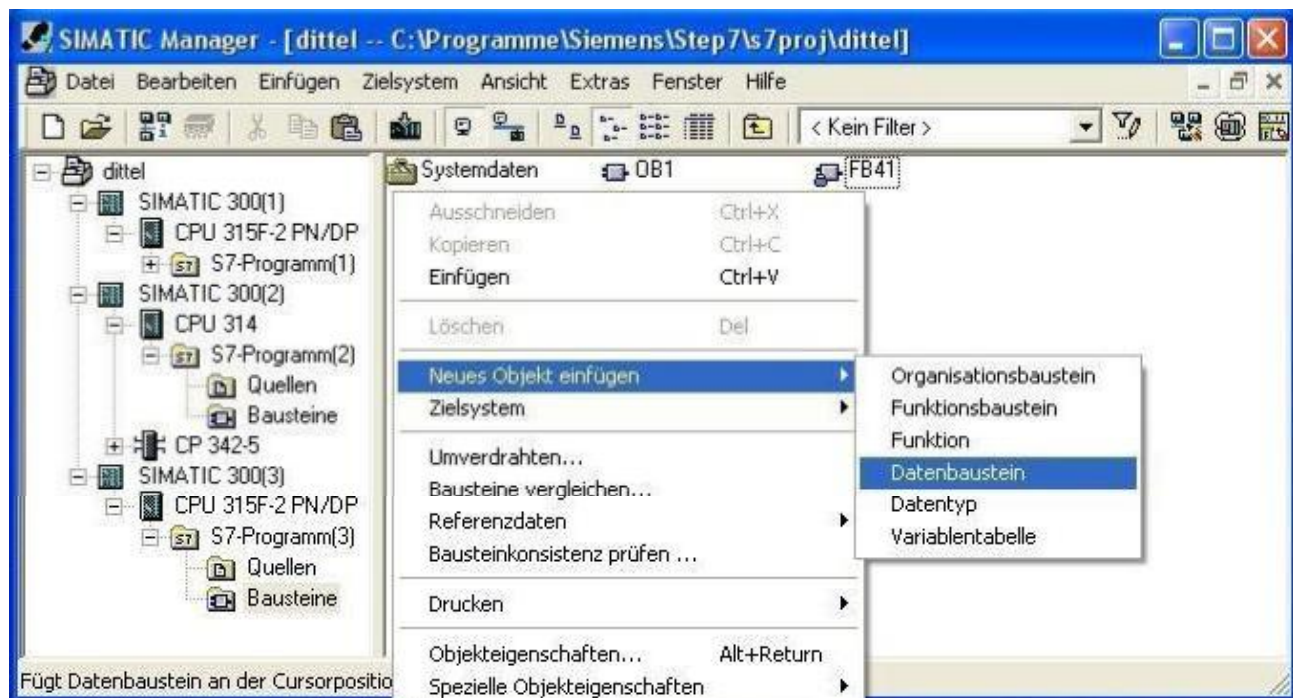


Abbildung 21: Datenbaustein DB41 einfügen

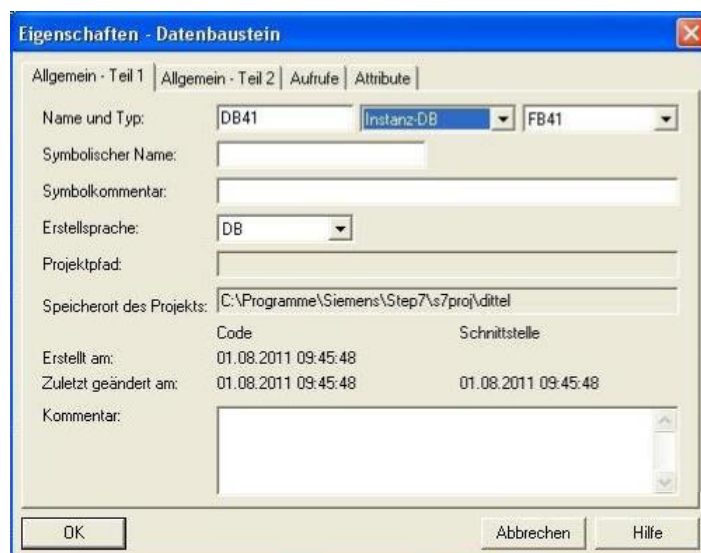


Abbildung 22: DB 41 als Instanz-Datenbaustein

Nach der Erzeugung sind keine weiteren Einstellungen notwendig. Es kann nun der OB 35 erzeugt werden (Kapitel 4.2.1.2 OB 35 Seite 10).

4.2.3.2 Globaler-Datenbaustein

Ein globaler Datenbaustein wird benötigt, um alle Variablen für externe Zugriffe, freischalten zu können.

Es wird ein Datenbaustein eingefügt (Abbildung 21), danach aber als Globaler Datenbaustein definiert (Abbildung 23).

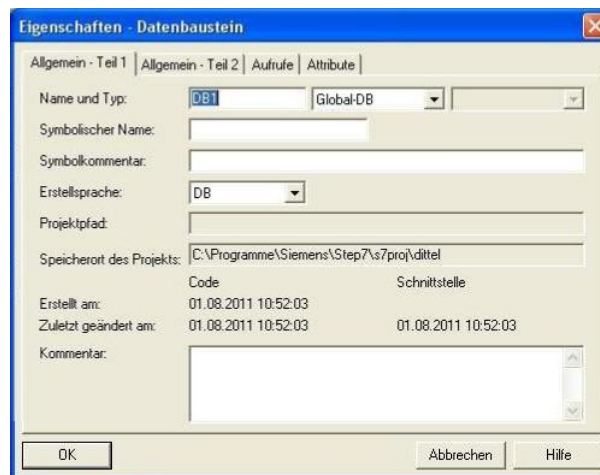


Abbildung 23: Globaler Datenbaustein

Alle notwendigen Variable, welche für die anderen Programme notwendig sind, werden im globalen Datenbaustein definiert.

Die erste, S7-300(1), Station enthält folgende wichtige Daten, welche global vorhanden sein müssen. Das ist die Variable "Temperatur" und "Heizwert", diese werden dem WinCC zur Verfügung gestellt. Der Baustein wird DB 1 genannt.

DB1 -- dittel\SIMATIC 300(1)\CPU 315F-2 PN/DP				
Adresse	Name	Typ	Anfangsw	Kommentar
0.0		STRUCT		
+0.0	Temperatur	INT	0	vorläufige Platzhaltervariable
+2.0	Heizwert	INT	0	
=4.0		END_STRUCT		

Abbildung 24: Datenbaustein DB 1

Ähnlich sind die Datenbaustein DB 2 und DB 3 aufgebaut. Eine unterschiedliche Datenbausteinbezeichnung, erleichtert den späteren Zugriff, von anderen Systemen.

Die Adressierung der Datenbaustein DB 2 und DB 3 werden in Tabelle 3 dargestellt.

DB	Adresse	Name	Typ	Anfangswert
DB 2	0.0	Sollwert2	INT	0
	2.0	Istwert2	INT	0
DB 3	0.0	Sollwert3	INT	0
	2.0	Istwert3	INT	0

Tabelle 3: globale Datenbausteine

Die Projektierung und Konfiguration ist fertig, nun werden die Bausteine in die SPS's geladen.

4.3 Zusammenfassung

Abschließend zu diesem Kapitel 4, werden die verschiedene Werte, zur besseren Übersicht, zusammengefasst.

Tabelle 4 dient zur Vereinfachung.

SPS-Nr	MPI-Adresse	Regelstrecke	DB-Nr.
S7-300(1)	3	Temperaturregelung	1
S7-300(2)	2,4	Füllstandsregelung mit Analogwertvorgabe	2
S7-300(3)	5	Füllstandsregelung mit WinCCwertvorgabe	3

Tabelle 4: Zusammenfassung SPS

5 Datenbank

In der Industrie nimmt die Verwaltung von Datenbanken steigend zu. Sie gewinnt immer mehr an Bedeutung.

Um Daten zwischen zu speichern und archivieren zu können, werden gern Datenbanken genutzt. Sie sind schnell zu programmieren und Zugriffe können nahezu zeitgleich erfolgen.

Es gibt verschiedene Datenbanktypen. Die Hochschule Mittweida benutzt, bevorzugt, den MS SQL Server 2005.

Ein externer Datenbankserver stand anfänglich zur Verfügung. Da dieser später defekt war, wurde auf den Microsoft SQL Server 2005 umgestiegen. Dieser SQL Server befindet sich direkt auf dem Arbeitsrechner.

5.1 Datenbank erzeugen

Nach dem Programmstart wird der Servername angezeigt. Dieser ist zu akzeptieren (Connect) bzw. zu ändern.

Über Databases / rechter Mausklick, den Menüpunkt New Databases... wählen.

Den Datenbanknamen eingeben, hier WinCCInTouch, und dann mit OK bestätigen.

Das Programm ist so überschaubar, das eine Datenbank mit mehreren Tabellen ausreichend ist.

5.2 Datenbanktabellen

Die Erstellung einer Datenbanktabelle erfolgt über WinCCInTouch / Tables und dann rechter Mausklick auf New Table... .

Die Spaltennamen und die dazugehörigen Datentypen werden dann direkt eingegeben. Nach Fertigstellung wird die Tabelle noch gespeichert.

Es gibt insgesamt fünf Tabellen. Drei Tabellen dienen zur Datenablage der SPS-Werte, sowie die Messprotokollnummer (MPN) und einer Identifikation (ID). In einer Tabelle wird die Messprotokollnummer aktualisiert. In der letzten Tabelle werden die MPN und Uhrzeiten mit Datum abgespeichert.

Die ersten drei Tabellen Fuell1, Fuell2 und Temperatur sollen die Werte der SPS der jeweiligen Regelung enthalten.

Die Tabellendefinitionen sind wie folgt festgelegt (Tabelle 5).

Fuell1 für analoge Wertvorgabe		Fuell2 für WinCCwertvorgabe		Temperatur	
ID	bigint	ID	bigint	ID	bigint
MPN	int	MPN	int	MPN	int
Istwert	int	Istwert	int	Temperatur	int
Sollwert	int	Sollwert	int	Heizwert	int

Tabelle 5: SQL-Tabellen

Eine weitere Tabelle dient zur Aktualisierung der MPN.

In der Tabelle MPN wird der Startwert 0 eingetragen, sonst kommt es beim WinCC zu Unstimmigkeiten.

In der Tabelle MPN gibt es nur eine Spalte, MPN, welche den Datentyp Integer besitzt.

Die letzte Tabelle Zeiten, dient zur Ablage der Startzeit, dem Startdatum, der Endzeit und dem Enddatum. Es wird jeder Messung eine MPN zugeordnet. Die Tabellendefinition ist in Tabelle 6 abgebildet.

Spaltenname	Datentyp
ID	bigint
MPN	int
Startzeit	char(10)
Startdatum	char(10)
Endzeit	char(10)
Enddatum	char(10)

Tabelle 6: SQL-Tabelle Zeiten

Die Wahl der Datentypen wird in Kapitel 5.3 Seite 19 näher erklärt.

Eine Besonderheit bei den Tabellen ist die Spalte ID. Sie dient zur besseren Kontrolle des Benutzers. Ein zählen wird damit vereinfacht.

Der Spaltenname ist ID und als Datentyp wird bigint gewählt. In den Spalteneigenschaften ändert man (Is Identity) von No auf Yes. Man kann auch die Wertsteigerung erhöhen, hier bleibt 1. In Abbildung 25 ist die Einstellung zu erkennen und in Abbildung 26 am Beispiel Füllstandsregelung 1 dargestellt.

Table - dbo.Fuell1			
Summary			
Column Name	Data Type	Allow Nulls	
ID	bigint	<input type="checkbox"/>	
MPN	int	<input checked="" type="checkbox"/>	
Istwert	int	<input checked="" type="checkbox"/>	
Sollwert	int	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

Column Properties	
Has Non-SQL Server Subscriber	No
Identity Specification	Yes
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1
Indexable	Yes
Merge published	No
(Is Identity)	

Abbildung 25: SQL ID

Table - dbo.Fuell1				
Summary				
ID	MPN	Istwert	Sollwert	
1	0	0	0	
2	0	0	0	
3	0	31	31	
4	27	31	31	
5	0	31	31	
6	0	31	31	
7	28	31	31	
8	28	30	31	
9	28	31	31	
10	28	30	31	
11	28	31	31	
12	28	30	8	

Abbildung 26: SQL ID Bsp.

5.3 Datentypen

Der SQL Server bietet viele Datentypen. Es kommen aber nur drei gültige Datentypen in Frage (Tabelle 7), da InTouch nur diese unterstützt.

Datentyp	Länge	Bereich	Variablentyp
int		-2.147.483.648 bis +2.147.483.648	Integer
float	15 Stellen	-1.79 E+308 bis +1.78 E+308	Real
char	8000 Zeichen		Meldung

Tabelle 7: SQL Datentypen [3]

5.4 Tabelleninhalte

Zum Testen des InTouch Projektes wurden die Tabellen bereits, mit SPS-Werten, gefüllt. Tabelle 8 zeigt die eingetragenen Daten.

Damit kann kontrolliert werden, ob die Anzeigefelder richtig funktionieren.

MPN	Startzeit	Startdatum	Messreihe	Anzahl Messwerte
27	14:23:21	06.09.11	Füllstandsregelung 1	1
28	14:26:16	06.09.11	Füllstandsregelung 1	17
29	14:26:47	06.09.11	Temperaturregelung 1	34
30	14:27:29	06.09.11	Füllstandsregelung 2	3
31	09:20:23	06.09.11	Füllstandsregelung 1	2

Tabelle 8: Datenbank Tabelleninhalte

6 WinCC

WinCC wurde als Bedienoberfläche gewählt, da bereits Erfahrung mit dieser Software gesammelt wurde. WinCC verfügt über die zwei Programmiersprachen C und Visual Basic Script (VBS). Es verfügt auch über eine Bibliothek mit sehr vielen vorgefertigten Elementen.

Die einzelnen Elementeerstellungen werden in der Bachelorarbeit "WinCC-Oberfläche mit Datenbankbindung und SPS-Anbindung für Medienuntersuchung im Grenzfall" näher erklärt. In den nachfolgenden Abschnitten wird nur auf die Bilder allgemein eingegangen.

Die Einstellung der einzelnen Button, E/A-Felder und Siemens Human-Machine-Interface (HMI) Symbole werden nur erwähnt. Sie können in der Bachelorarbeit, ab Kapitel 3.3.2. Seite 21, nachgelesen werden.

Es wird ein Startfenster geöffnet, in dem der Benutzer die gewünschte Regelung auswählen kann. Es kann zwischen einer Füllstandsregelung mit Analogwertvorgabe, einer Füllstandsregelung mit Sollwerteingabe und einer Temperaturregelung gewählt werden. Bei der gewünschten Regelung angekommen, werden die aktuellen Sollwerte und Istwerte angezeigt und die Messung können gestartet bzw. beendet werden. Ein Wechsel zum Startbild ist bei jeder Messung möglich. Es wurde eine Benutzerumgebung geschaffen, die bedienerfreundlich ist.

6.1 Variablen

Im WinCC werden interne und globale Variablen benötigt. Die verwendeten globalen Variablen wurden in der SPS in den jeweiligen globalen Datenbausteinen (Kapitel 4.2.3.2 Seite 14) definiert.

Der Tabelle 9 können die internen Variablen entnommen werden und in Tabelle 10 die globalen Variablen. MPI kennzeichnet die MPI-Adresse (Parameter).

Die internen Variablen gehören der Variablengruppe SPS an.

Die Variablenverbindung zur SPS wird in der Bachelorarbeit Kapitel 3.3.1.2 ab Seite 18 näher erklärt. Die verwendeten Treiber werden vorausgesetzt.

Name	Typ
beenden	Binäre Variable
edatum	Textvariable 16-Bit Zeichensatz
ezeit	Textvariable 16-Bit Zeichensatz
mpn	Vorzeichenloser 8-Bit Wert
sdatum	Textvariable 16-Bit Zeichensatz

Name	Typ
starten	Binäre Variable
szeit	Textvariable 16-Bit Zeichensatz

Tabelle 9: WinCC interne Variablen

Verbindung	MPI	Name	Typ	Parameter
Füllstand1	MPI 2	Sollwert1	Vorzeichenloser 16-Bit Wert	DB2,DW0
		Istwert1	Vorzeichenloser 16-Bit Wert	DB2,DW2
Füllstand2	MPI 5	Sollwert2	Vorzeichenloser 16-Bit Wert	DB3,DW0
		Istwert2	Vorzeichenloser 16-Bit Wert	DB3,DW2
Temperatur	MPI 3	Temperatur	Vorzeichenloser 16-Bit Wert	DB1,DW0
		Heizwert	Vorzeichenloser 16-Bit Wert	DB1,DW2

Tabelle 10: WinCC globale Variablen

6.2 Startbild

Im Startbild, Start.pdl, findet eine Auswahl zur Messreihe statt. Die einzelnen Buttons sind Bildwechselbuttons. Ein Beenden-Button beendet den Runtime-Prozess.

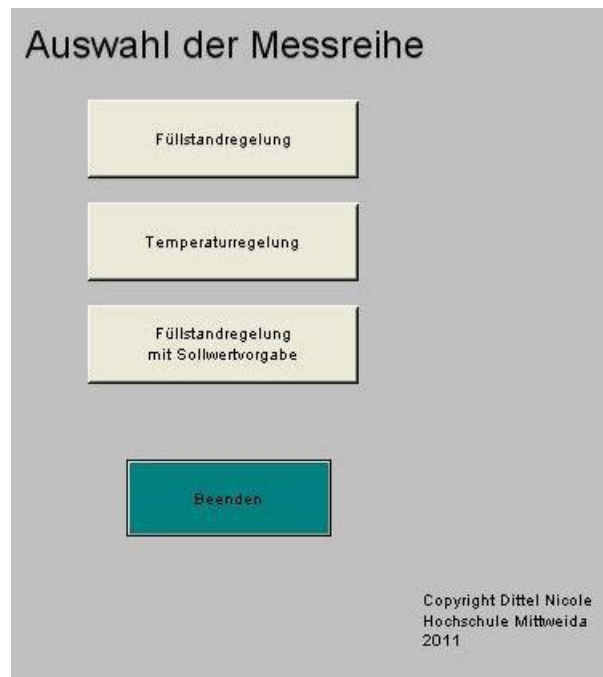


Abbildung 27: WinCC Startbild

Das Bild Start.Pdl wird als Graphics-Runtime-Startbild gewählt.

6.3 Füllstandsregelung 1

Im Füllstandsregelbild 1, Fuell1.Pdl, wird der Sollwert, sowie der Istwert der SPS entnommen. Der Istwert wird im Tank als Füllanzeige angezeigt.

Der Button "Messung starten" dient zum Starten einer Messung, sobald dieser gedrückt wurde, wird nur noch der "Messung beenden" Button angezeigt. Der "andere Messung" Button wird nicht angezeigt, dadurch entsteht keine Abbruch des Schreibens in die Datenbank. Den "Messung beenden" Button betätigt man, wenn die Messung beendet werden soll.

Wenn der "Messung beenden" Button gedrückt wurden wird dieser nicht mehr angezeigt und die Button "Messung starten" und "andere Messung" erscheinen.

Der "andere Messung" Button ist ein Button mit Fensterwechselfunktion zum Bild Start.

Die Beschreibung (links) kennzeichnet den Standort der Messreihe, hier linke Messreihe.

Die verwendeten Variablen sind "Sollwert1", "Istwert1" als Ausgabevariablen und "starten" bzw. "beenden" als dynamische Wertebereichsanzeige für die Buttons. Die Quelltexte sind im Anhang ab Seite B nachzulesen.

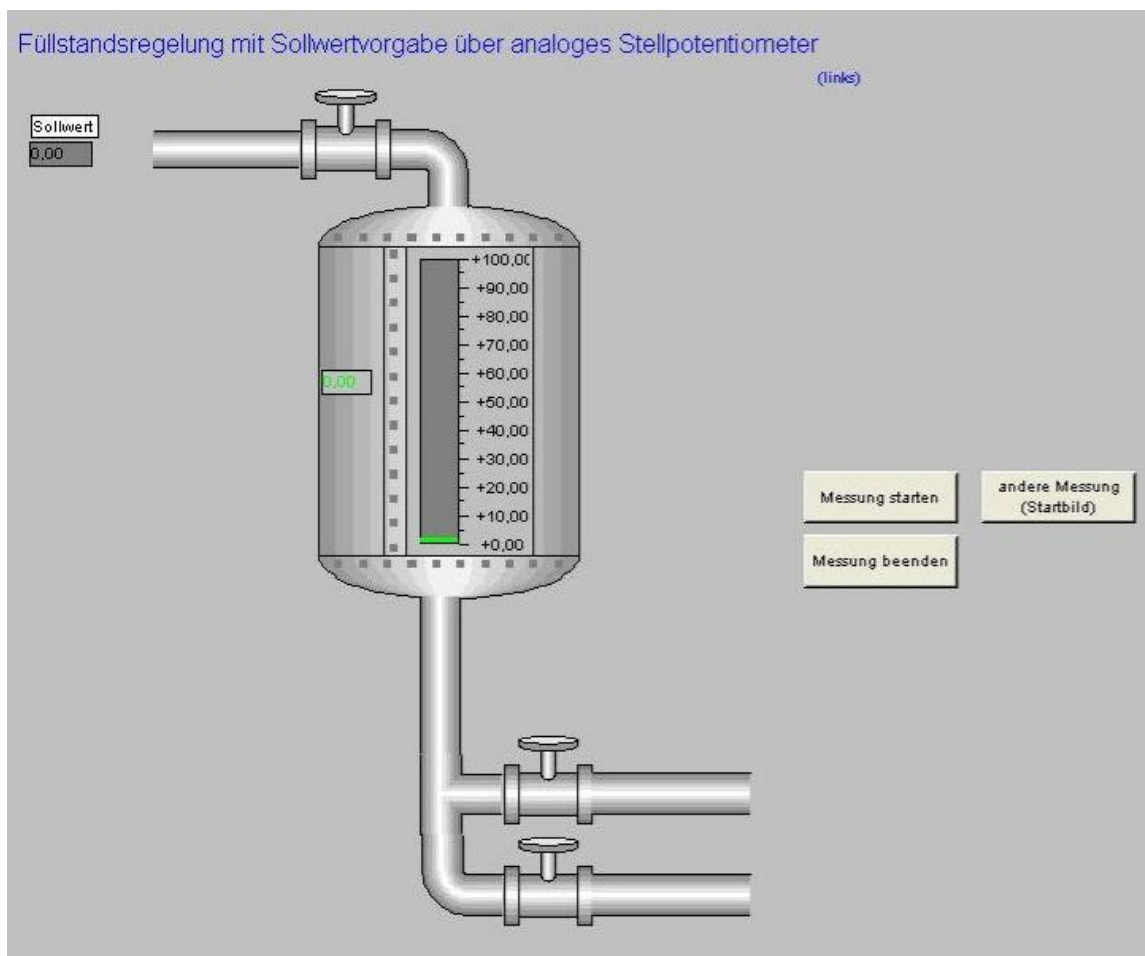


Abbildung 28: WinCC Füllstandsregelung 1

6.4 Füllstandsregelung 2

Die zweite Füllstandsregelung ist analog der ersten, bis auf die Besonderheit, dass der Sollwert im WinCC eingetragen und an die SPS übergeben wird.

Die hier verwendeten Variablen sind "Sollwert2" als Eingabevariable und "Istwert2" als Ausgabevariable.

Der Quelltext ändert sich in den Zeilen des Datenbankzugriffes. Es wird nicht auf die Datenbanktabelle Fuell1, sondern Fuell2 zugegriffen (Änderung nur beim "Messung beenden" Button, Quelltextseite D). Diese Messreihe befindet sich rechts.

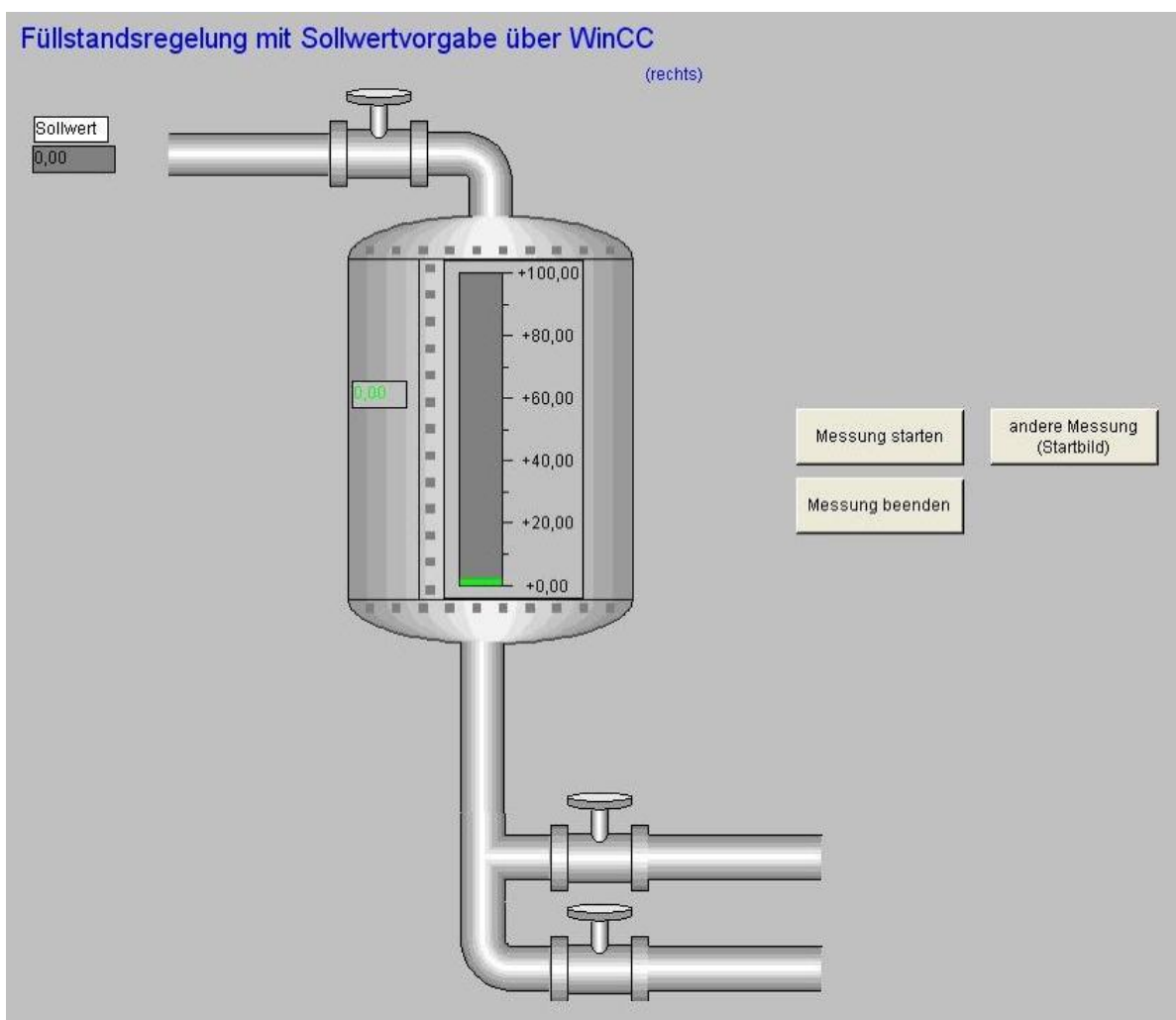


Abbildung 29: WinCC Füllstandsregelung 2

6.5 Temperaturregelung

In dem Bild Temperaturregelung, Temp.Pdl, wird der aktuelle SPS-Wert als Digitalwert angezeigt. Zur Temperaturanzeige wurde ein Rechteck eingefügt (Rechteck2), mit der Füllverknüpfung zur Variablen "Heizwert".

Der Quelltext ändert sich in den Zeilen des Datenbankzugriffes. Es wird nicht auf die Datenbanktabelle Fuell1, sondern Temperatur zugriffen (Änderung nur beim "Messung beenden" Button, Quelltextseite D). Zu Beachten sind die Änderungen der Variablen Diese Messreihe befindet sich in der Mitte.

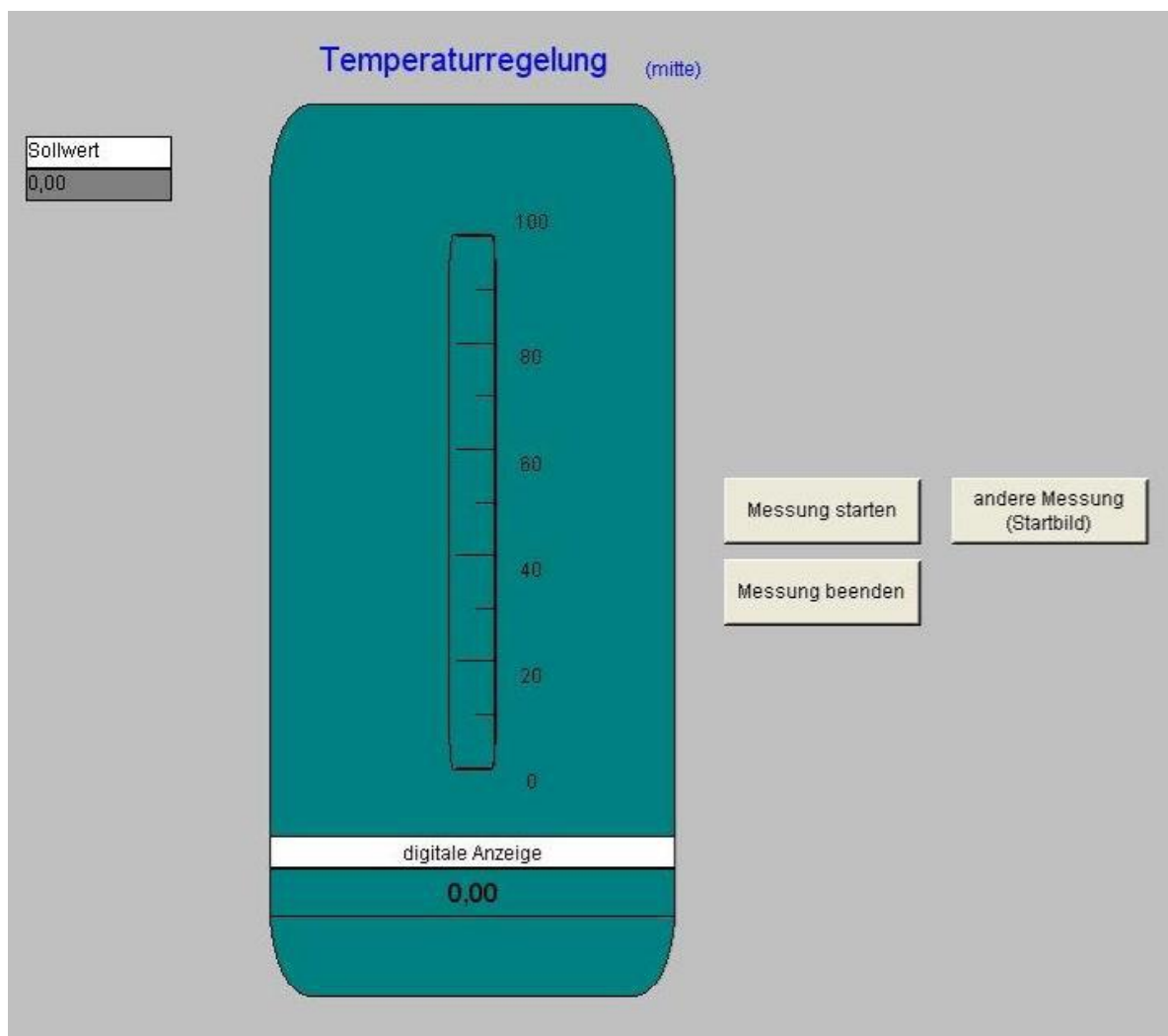


Abbildung 30: WinCC Temperaturbild

7 InTouch

Das Programm InTouch wurde von der Firma Wonderware entwickelt und ist ideal dazu geeignet, um eine Visualisierung zu realisieren.

Diese HMI-Software bietet zahlreiche Möglichkeiten zur besseren Projektentwicklung. Das Menü ist sehr bedienerfreundlich angelegt. Es bietet die Möglichkeit eines dezentralen Alarmings in Echtzeit mit Verlaufsanzeigen zu Analysezwecken. [5]

Das InTouch Projekt soll Daten aus der Datenbank lesen.

Dies wurde so realisiert, dass verschiedene Abfragen gestellt werden und eine Suchfunktion den richtigen Datensatz anzeigt. Es wird immer nur ein Datensatz angezeigt und kann über Buttons (Next und Prev) den nächsten oder vorherigen Datensatz auswählen.

7.1 InTouch-Projekt erstellen

Um InTouch zu benutzen, ist es notwendig ein neues Projekt anzulegen.

Den rot markierten Menüpunkt, in Abbildung 31 dient zur Erzeugung eines neuen Projektes.

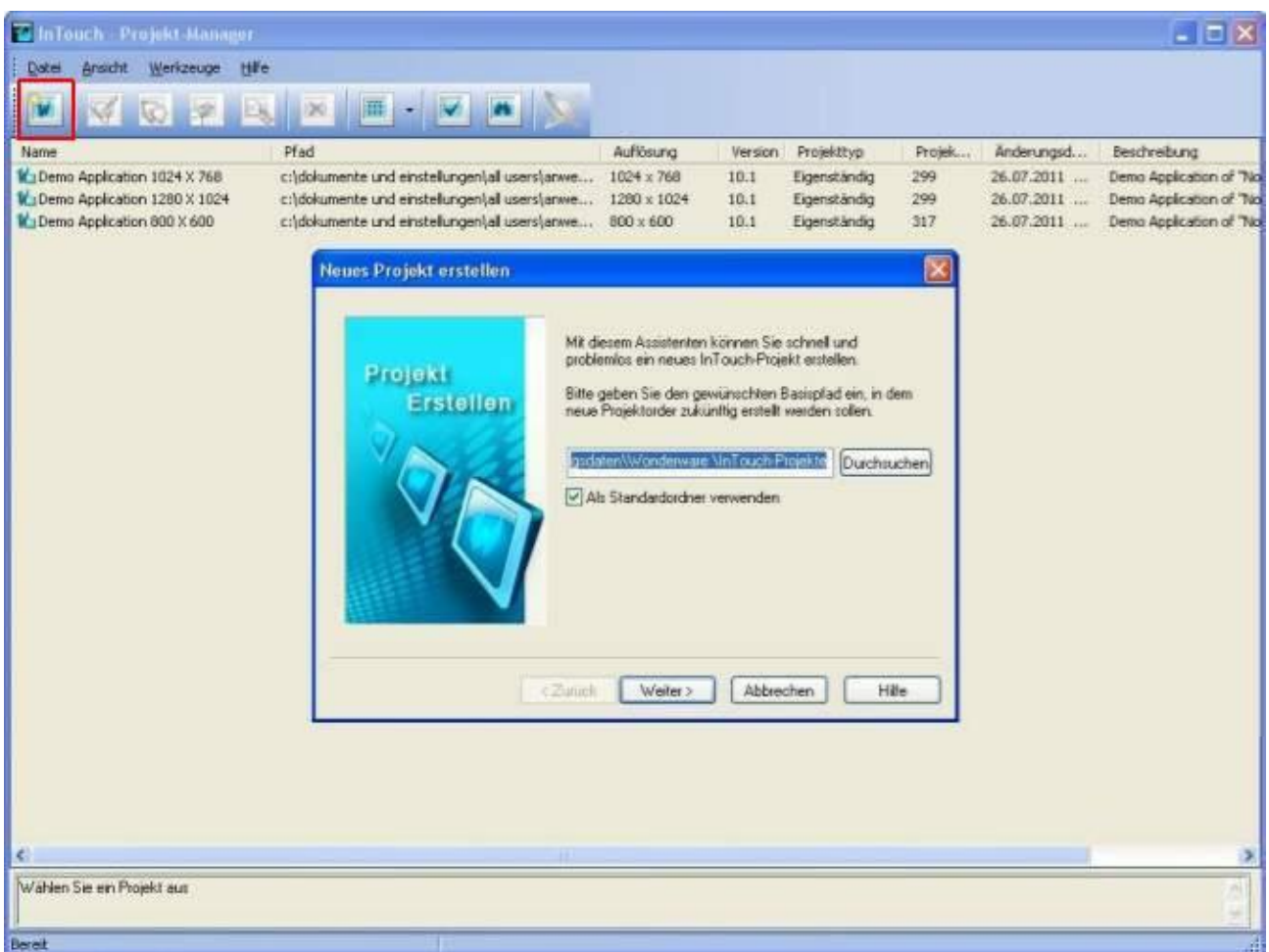


Abbildung 31: InTouch Projektierung

Der Ordner, in dem sich das Projekt befindet, heißt neuesproj (Abbildung 32). Die Änderung des Ordernamens wird in Kapitel 9.2.1 Seite 40 näher erklärt.



Abbildung 32: InTouch Ordnername

Als Projektname wurde Verbindung gewählt (Abbildung 33). Da es zu Komplikationen kam wurde später ein neues Projekt, mit dem Namen, Neues InTouch-Projekt. Die Fehlerbeschreibung wird in Kapitel 9.2.1 Seite 40 näher erläutert. Es kann dazu kommen, das Bilder den alten Projektamen enthalten.



Abbildung 33: InTouch Projektname

Nachdem alle Einstellungen getroffen und bestätigt wurden, erscheint das Projektionsfenster Abbildung 34.

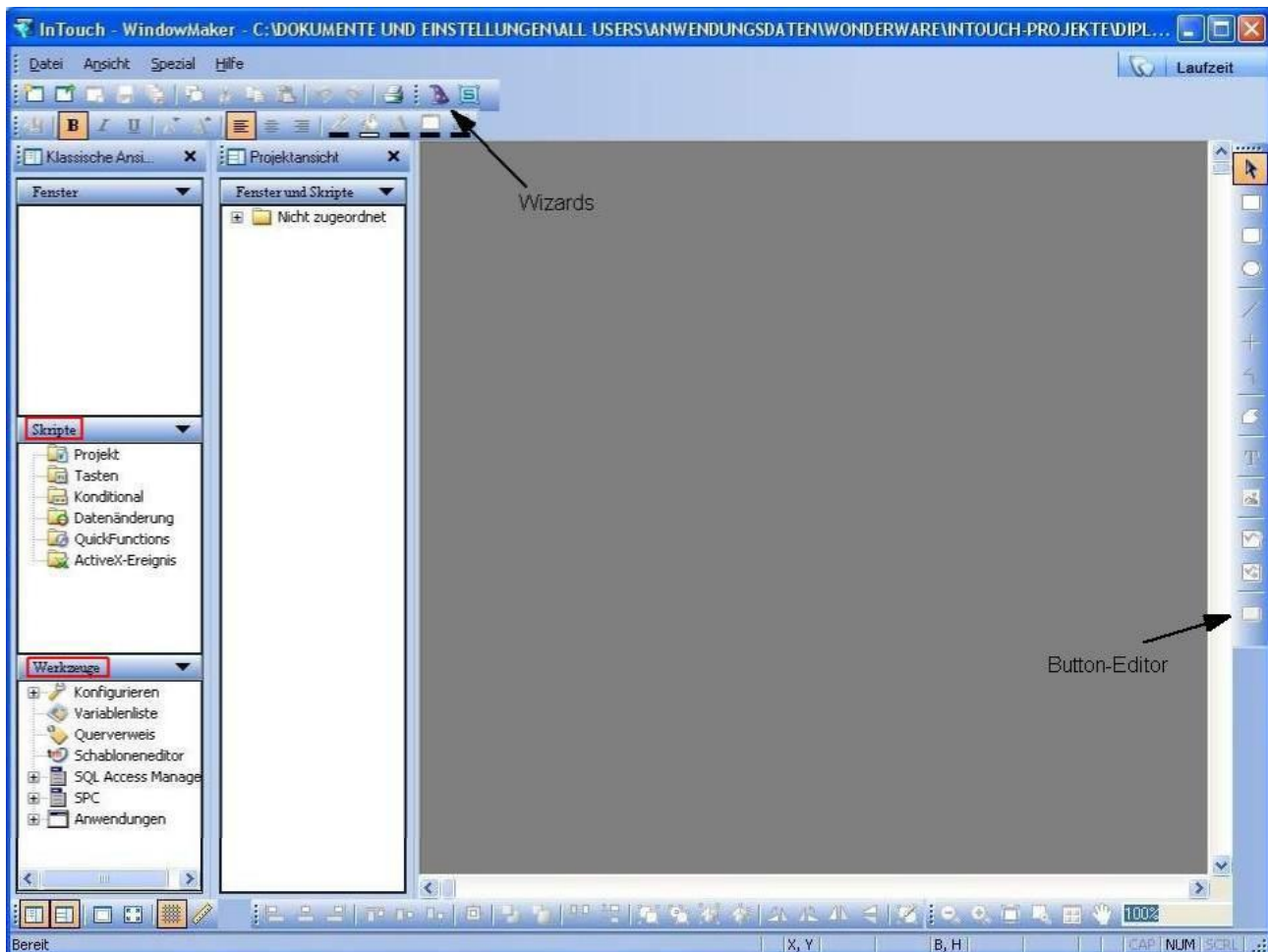


Abbildung 34: InTouch Fenster

Es befinden sich in der Menüleiste verschiedene Button, u.a. Wizards. Er dient als Assistent, zum Einfügen vorgegebener Symbole. Diese Symbole sind unter anderem Meldfenster, Diagramme, Displays, Schieberegler, Windows-Steuerelemente und auch Runtime Tools.

Im linken Teil des WindowMaker befinden sich Fenster, wie Werkzeuge, Skripte und Fenster. In Fenster werden alle erzeugten Fenster angezeigt, es erleichtert das Öffnen und Schließen der einzelnen Fenster. Wenn man im Fenster Skripte, einen Menüpunkt auswählt, kommt man direkt zur Skriptprogrammierung. Das letzte Fenster heißt Werkzeuge und enthält alle wichtigen Funktionen für die SQL-Datenbankbeschreibung und die Statische Prozeßlenkung (SPC) - Datenbankbeschreibung.

In Abbildung 35 wird das Grundkonzept der InTouch-Regelung dargestellt.

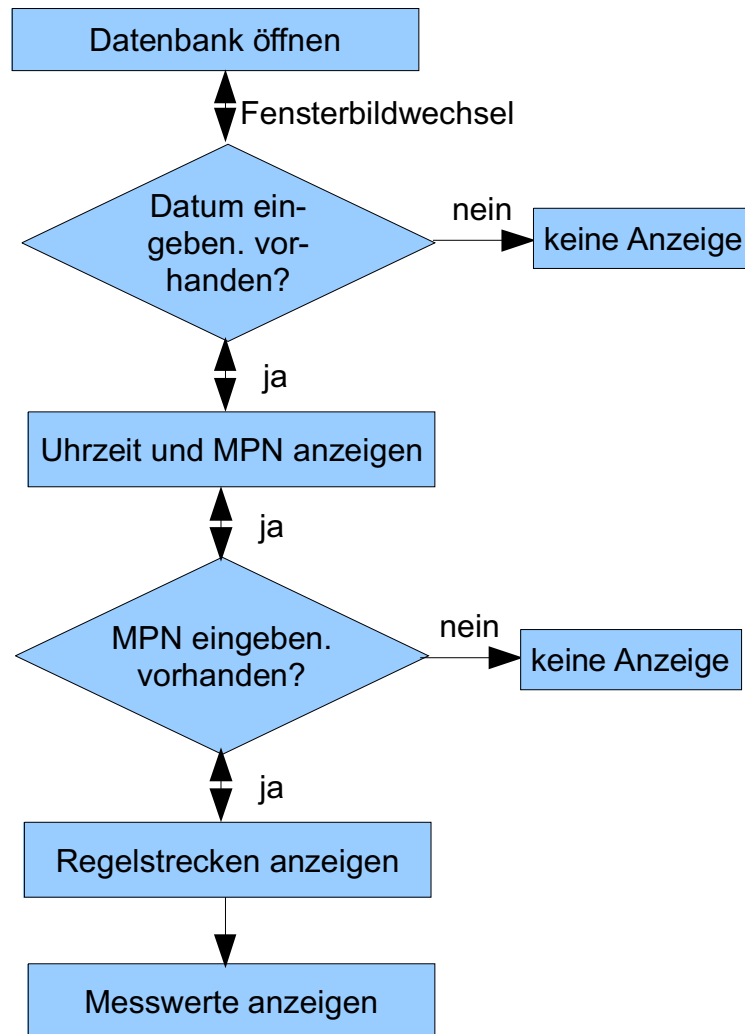


Abbildung 35: InTouch

7.2 Neues Fenster

Die Projektierung kann nun erfolgen.

Dazu wird ein neues Fenster angelegt (Datei / Neues Fenster). Name, Kommentar und Fensterfarbe und andere Einstellungen eingeben und das neue Fenster wird erzeugt (Abbildung 36).



Abbildung 36: InTouch Fenstereigenschaften

Die Größe des Fenster ist zu einem späteren Zeitpunkt auch noch änderbar.

Insgesamt werden fünf Fenster konstruiert.

Das Startbild zeigt ein Text an und enthält einen Bildwechselbutton mit Datenbankanbindung.

Dieses Suchfensterbild zeigt, nach vielen Benutzereingaben, die Messreihe an, in der die gewünschten Datensätze vorhanden sind.

Die restliche drei Fenster sind die einzelnen Bilder der Datensatzanzeige der verschiedenen Messreihen.

7.3 Variablen erstellen

Ohne Variablen läuft nichts, deshalb werden folgende Variablen, zusätzlich zu den Systemvariablen, angelegt (Tabelle 11).

Dies geschieht über Spezial / Variablenliste. In der Variablenliste kann später der Datentyp geändert werden, aber nur die Größe des zuvor festgelegten Datentyps.

Variablenname	benötigt für	Datentyp
anlage	enthält Text für Messreihenauswahl	Speicher Meldung
Anzahl	Anzahl der ausgewählten Datensätze	Speicher Integer
anzeige	Sichbarkeitsdarstellung oberer Teil	Speicher Binär
anzeige2	Sichbarkeitsdarstellung unterer Teil	Speicher Binär
Datum	Startdatum aus Datenbank	E/A Meldung
f1	Zeilenanzahl der Datenbanktabelle Fuell1	Speicher Integer
f2	Zeilenanzahl der Datenbanktabelle Fuell2	Speicher Integer
f3	Zeilenanzahl der Datenbanktabelle Temp.	Speicher Integer
Heizwert	Heizwert aus Datenbank	E/A Integer
IDNr	Verbindungs-ID	Speicher Integer
Istwert	Istwert aus Datenbank	E/A Integer
MPN	MPN aus der Datenbank	Speicher Integer
MPN_int	eingetragene MPN	Speicher Integer
RESULTCODE	Platzhalter für Fehlermeldung	Speicher Integer
Sollwert	Sollwert aus Datenbank	E/A Integer
Temperatur	Temperatur aus Datenbank	E/A Integer
Uhrzeit	Benutzereingabe Startzeit	Speicher Meldung
Uhrzeit_DB	Startzeit aus Datenbank	E/A Meldung

Tabelle 11: InTouchvariable

Zu Beachten ist der Datentyp E/A ... und Speicher

Speichervariablen haben keine Verbindung zur "Außenwelt". Sie eignen sich für Konstanten und Variablen, die ausschließlich innerhalb des InTouch-Projektes benötigt werden. Es können auch Berechnungsergebnisse abgelegt werden, auf die andere Programme zugreifen können. [6]

E/A-Variablen sind mit einer externen Datenquelle verknüpft. Dies kann zum Beispiel eine SPS, ein Prozessrechner oder ein anderer Rechner im Netzwerk sein. [6]

7.4 Variablen löschen

Zu erwähnen wäre auch, wie man nicht verwendete Variablen (Testvariablen) wieder löschen kann. Nachfolgend wird die Vorgehensweise Schritt für Schritt erklärt.

Zuerst müssen alle Fenster geschlossen sein. Dann geht man über den Menüpunkt Spezial und wählt Punkt Benutzungszähler aktualisieren. Nun werden alle Verbindungen zwischen Variablen und Skripten überprüft und alle nicht verwendeten Variablen aussortiert.

Wurde diese Aktion erfolgreich ausgeführt wählt man nun den Menüpunkt Nicht benötigte Variablen löschen in Menü Spezial aus. Jetzt wählt man die zu löschenden Variablen aus und bestätigt zweimal mit OK.

7.5 Bindeliste

Wichtig für eine SQL-Verbindung ist eine Bindeliste. Es ist notwendig, dass jeder Datenbanktabellen, welche verschiedene Spaltennamen besitzt, eine eigene Bindeliste erhält.

Es wird eine Variablenzuweisung erstellt.

Um eine neue Bindeliste zu erstellen werden folgende Menüpunkte ausgeführt. Spezial / SQL Access Manager / Bindeliste oder über die Werkzeuge (linker Bildschirmrand) / SQL Access Manager / Bindeliste.

Ist noch keine Bindeliste vorhanden, muss eine neue Bindeliste erstellt werden. Zuerst wird ein sinnvoller Name gewählt, hier Zeiten (Abbildung 37). Danach werden die Variablen den Spaltennamen zugeordnet und hinzugefügt.

Durch anklicken des Variable Feldname Feld erscheint ein Variablenauswahlmenü. Das Eintragen der vorhandenen Variablen vereinfacht sich.



Abbildung 37: Bindeliste konfigurieren

Folgende Variablen müssen zugeordnet werden.

Bindelistenname	Variable	Spaltenname
Fuell1	MPN	MPN
	Istwert	Istwert
	Sollwert	Sollwert
Temperatur	MPN	MPN
	Temperatur	Temperatur
	Heizwert	Heizwert
Zeiten	MPN	MPN
	Datum	Startdatum
	Uhrzeit_DB	Startzeit

Tabelle 12: InTouch Bindelisten

7.6 WindowMaker Fenster

In diesem Kapitel wird die genaue Fensterbeschreibung dargestellt.

Wie bereits erwähnt gibt es fünf Fenster. Das erste Fenster ist das Startfenster "Start".

Vorab noch einige Informationen. Das #-Symbol wurde für Textein- bzw. ausgaben verwendet. Das Symbol verschwindet wenn der Benutzer eine Eingabe tätigt. Steht ein vordefinierter Text im Feld, bleibt dieser Text erhalten und der Variablenwert wird hinzugefügt.

Beendet wird der Runtime über den Menüpunkt Datei / Beenden.

Das "Datenbank öffnen" Skript darf nicht im Fensterskript programmiert werden, durch die Zeiteingabe, aller wieviel Sekunden wiederholen, wird die Verbindungs-ID immer um eins erhöht und die anderen SQL-Funktionen, wie SQLPrev und SQLNext können nicht ihren Zweck erfüllen (Kapitel 9.2 Seite 40). Das "Datenbank öffnen" Skript wird auf einem Button hinterlegt. Deshalb wurde auch ein Startfenster eingefügt.

Das Kopieren von Elementen kann mit STRG + C durchgeführt werden. Im Fenster wird das zu kopierende Element ausgewählt und mit STRG + C in die Zwischenablage verschoben. Nun wird in dem gewünschten Fenster geöffnet und STRG + V gedrückt. Es wandelt sich, nach einer Weile, der Mauszeiger in ein kleines rechtwinkliges Winkelsymbol um. Es wird nun die Stellen zum Einfügen gesucht und angeklickt. Das Element wird so eingefügt, dass das kleine Symbol, die linke obere Ecke des Elementes, ist.

7.6.1 Start

Ein Startbild zeigt einen Begrüßungstext und enthält einen Button (Abbildung 38). Der Button startet die Datensatzanzeige, es wird das Fenster Suchfenster geöffnet und intern wird die Datenbank WinCCInTouch geöffnet (Quelltext Seite E).

Ein Bitmap wurde über den Button in der rechten Symbolleiste erzeugt. Es wurde ein Bild vom Typ jpg eingefügt.



Abbildung 38: InTouch Startbild

7.6.2 Suchfenster

In diesem Bild werden Daten vom Benutzer abgefragt und Daten aus der Datenbank angefordert (Abbildung 40).

Es wurde ein Fensterskript integriert. Im Fenster mit der rechten Maustaste klicken und Fensterskripte auswählen. Der Quelltext befindet sich auf Seite E. Folgende Einstellungen sind zu treffen, Solange angezeigt und Alle 1ms (gewünschte Zeit eintragen). Sobald ein Datenbanksatz gefunden wurde, werden weitere Elemente zu Ansicht freigegeben. Bei positiver Suche wird eine binäre Variable eins gesetzt.

Der Platzhalter `##.##.####` dient dem Eintrag der Variable "Datum" und wird über den Button Text aus der linken Symbolleiste erzeugt. Kennzeichnung eintragen und doppelklick auf das Textfeld. Nun erscheint das Aktionsfenster (Abbildung 39). Es muss in der Benutzereingabe / Text die Variable "Datum" eingetragen werden.



Abbildung 39: InTouch Aktionsfenster

Es wurden zwei Rahmen um die Textfelder gelegt. Über den Assistenten / Rahmen / einfacher geprägter Rahmen war das ein schnelles Einfügen. Der Nachteil ist die Bedienbarkeit, eine Sichtbarkeitseinschränkung war nicht möglich. Die Rahmen werden in den Hintergrund gestellt.

Als nächstes folgt ein Button. Buttons werden über den Button-Editor (rechts in der Abbildung 34 Seite 27) erzeugt. Anklicken und den Button in seine gewünschte Größe ziehen und loslassen. Es wird der Button erzeugt. Nach einem Doppelklick erscheint wieder das Aktionsfenster (Abbildung 39).

Der Quelltext, des Button "Nach Datensätze suchen", wird in dem Menüpunkt Schaltflächen / Aktion hinterlegt. Es wird in der Datenbank Zeiten, mit Verbindung zur Bindeliste Zeiten, das Startdatum mit der Variable "Datum" verglichen und zwischengespeichert. Die genaue Zeilenanzahl wird ermittelt. Der Quelltext befindet sich im Anhang auf Seite E.

Alle nachfolgenden Elemente haben zusätzlich die Einschränkung, das die Sichtbarkeit von Variablen "anzeigen" abhängig ist.

Der Button "Prev" dient zur Anzeige des vorherigen Datensatzes. Der Button "Next" hingegen den nachfolgenden Datensatz (Quelltext Seite E und E).

Die Uhrzeit-Anzeige ist ein Meldefenster, erstellt mit dem Assistenten / Textanzeigen / Meldungsvariablen-Anzeige und verbunden mit der Variable "Uhrzeit_DB".

Der Platzhalter # ist verbunden mit der Variablen "MPN" im Menüpunkt Wertanzeige / Analog. Hier wird der Wert nur angezeigt, er kann nicht geändert werden.

Die Schaltfläche "gefundene Werte anzeigen" zeigt den ersten Datensatz der zwischengespeicherten Datenbanktabelle an (Quelltext Seite E).

Es gibt 0 Werte ist ein Text, bei dem die Null die Verbindung zur Variablen "Anzahl" aufweist. Die Wertanzeige ist Analog.

Das nächste # Symbol dient zur MPN Eingabe. Die Benutzereingabe ist Analog und die Variable "MPN_int" wird eingefügt.

Bei der Schaltfläche "Anlage anzeigen" findet ein Suchmechanismus statt. Es werden alle Datenbanktabellen mit der, vom Benutzer eingetragenen MPN, untersucht und nach Überstimmungen geprüft. D.h. es werden in die Variablen "f1", "f2" und "f3" die Anzahl der jeweiligen Zeilen aus der Datenbanktabelle eingetragen. Ist eine der Variablen größer Null, wird die Variable "anlage" beschrieben und die Bitvariable "anzeigen" wird gesetzt. Sollte keine MPN vorhanden sein wird eine Fehlermeldung angezeigt. Der Quelltext ist im Anhang auf Seite F.

Nun folgt wieder für alle nachfolgenden Elemente eine zusätzlich Einschränkung, der Sichtbarkeit der Variablen "anzeigen2".

In dem Textfenster "Messwerte der Anlage ####", wird der Wert der Variable "anlage" wiedergegeben. Wertanzeige ist Text, da die Variable eine Meldungvariable ist.

Die Schaltfläche "anzeigen" führt folgende Aktion aus. Ein Bildwechsel zu einem bestimmten Bild wird durchgeführt. Das jeweilige Bild ist abhängig von den Variablen "f1", "f2" und "f3" (Quelltext Seite F).

Der Zurück-Button wechselt in das Startfenster und setzt alle Variablen zurück (Quelltext Seite G).

The screenshot shows a window titled "Suchfenster" with a pink header asking "Welche Datensätze sollen angezeigt werden?". Below this, there is a search section with a label "Datum eintragen", a text input field containing "##.##.####", and a button "Nach Datensätzen suchen".

Below the search section, there is a table-like structure. On the left is a "Prev" button, followed by the text "Uhrzeiten" and "Meldung". In the center is a vertical separator line. To the right of the line is the text "MPN" and "#". On the far right is a "Next" button. Below this structure is a button labeled "gefundene Werte anzeigen".

Below the button, it says "Es gibt 0 Werte".

Below a horizontal line, there is a section for "gewünschte MPN eintragen" with a text input field containing "#", and a button "Anlage anzeigen".

Below another horizontal line, there is a section for "Messwerte der Anlage" with a button "anzeigen".

Below this, there are three large hash symbols "###".

At the bottom, there is a button labeled "Zurück".

Abbildung 40: InTouch Suchfensterbild

7.6.3 Regelungsfenster

Es gibt drei Regelungsfenster, Fuellanlage1(), Fuellanlage2 und Temperaturregelung.

Die eckigen Klammern [] verweisen auf die Änderungen der Variablenzuweisung hin.

Das Textfeld "So viele Werte gibt es #", ist eine Analog Wertanzeige mit der Variablen "f1" ["f2", "f3"], diese Variablen enthalten die Anzahl der Tabellenzeilen.

Die Schaltfläche erster Wert ruft den ersten Datensatzes auf (Quelltext Seite G).

Sollwert [Heizwert] # ist eine Analog Wertanzeige mit der Variablenverbindung "Sollwert" ["Heizwert"].

Analog verhält sind die Istwert-Anzeige ["Temperatur"]. Die Variable ist "Istwert" ["Temperatur"]

"Vorheriger Wert" zeigt den vorherigen Datensatz an. "Nächster Wert" zeigt den nächsten Datensatz (Der Quelltext ist gleich dem Prev- bzw. Next-Button Seite).

Die Schaltfläche Drucken druckt das gesamte, gerade geöffnete Fenster, aus. Der abgespeicherte Dateityp ist xps (Quelltext Seite F).

Der Zurück-Button ist gleich dem Zurück-Button im Suchfenster, nur das man hier ins Suchfenster und nicht Startfenster gelangt.

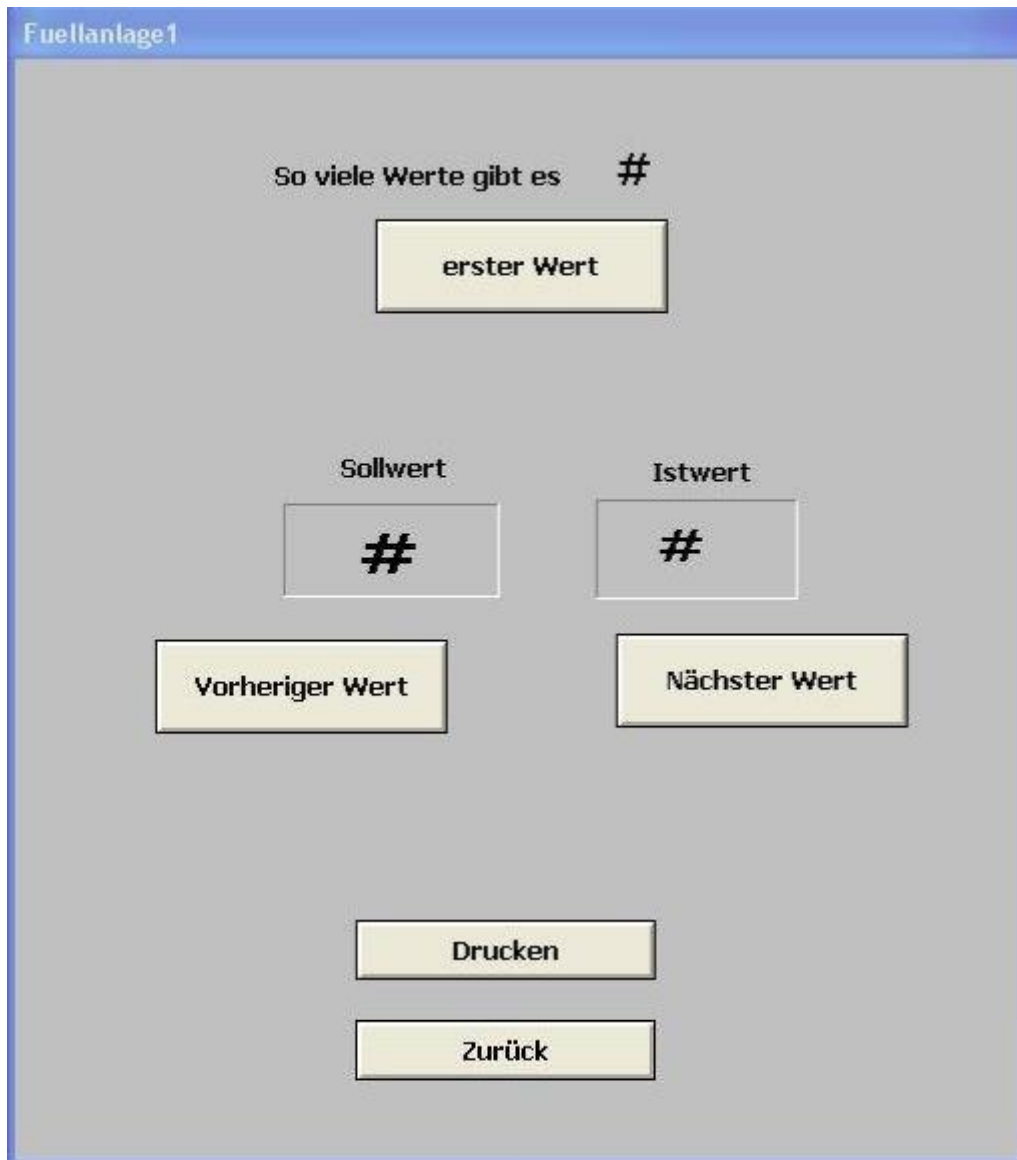


Abbildung 41: InTouch Füllanlage 1 Bild

8 Verbindung

Um mit der Datenbank arbeiten zu können, bedarf es einer Verbindung zwischen dem SQLServer und dem Client.

Es wurde mit einem externen Server begonnen. Da technische Problem auftraten wurde zu einer MS SQL Server Datenbank ausgewichen. Die Fehlerbeschreibung ist im Kapitel 9.3 auf Seite 41 nachzulesen.

8.1 ODBC-Datenquelle

Als Datenbankschnittstelle für den SQL-Server wurde der Open Database Connectivity (ODBC) verwendet.

Start \ Einstellungen \ Systemsteuerung \ Verwaltung \ Datenquellen (ODBC)

Als neue Datenquelle wird ein SQL-Server erstellt. Als Servername wurde intern zugewiesen. Der Servername ist wichtig für die Verbindung zwischen WinCC bzw. InTouch und dem SQL-Server. In Beschreibung wird nur ein Kommentar hinzugefügt.

Die Serverbezeichnung ist die Adresse des Rechners.

Die Standarddatenbank ist auf WinCCInTouch zu ändern.

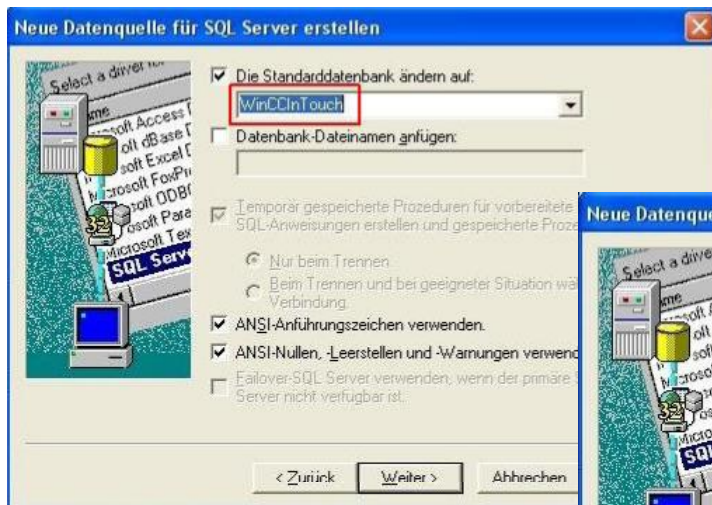


Abbildung 42: Standarddatenbank

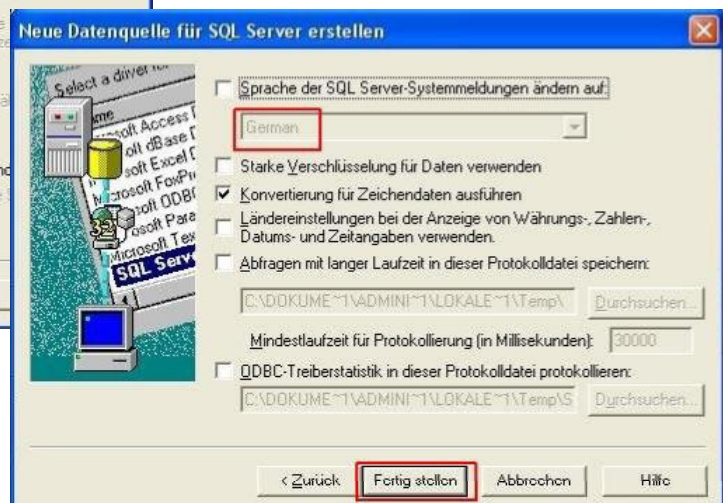


Abbildung 43: weitere Einstellungen

Nachdem die Standardsprache auf die gewünschte Sprache eingestellt wurde (hier German) kann die Konfiguration fertig gestellt werden.

8.2 Verbindungsaufbau

Die SQL-Datenquelle ist, nach Kapitel 8.1, fertig eingestellt. Nun kann die Datenquelle getestet werden (Abbildung 44).

Nach erfolgreichem Datenquellentest kann der Test mit OK beendet werden.

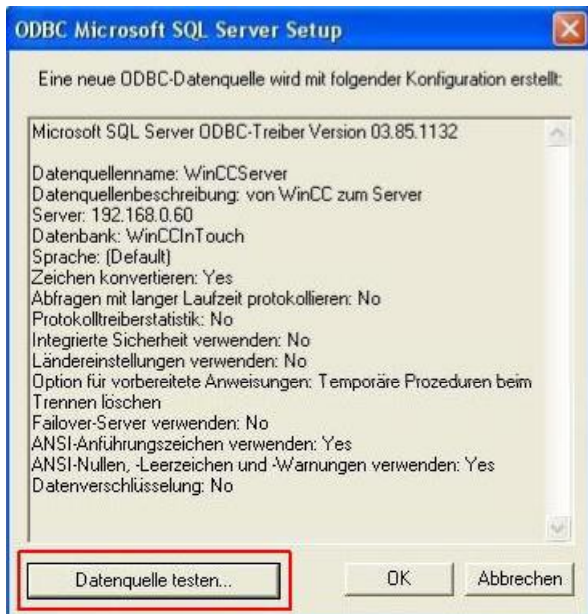


Abbildung 44: Datenquelle testen

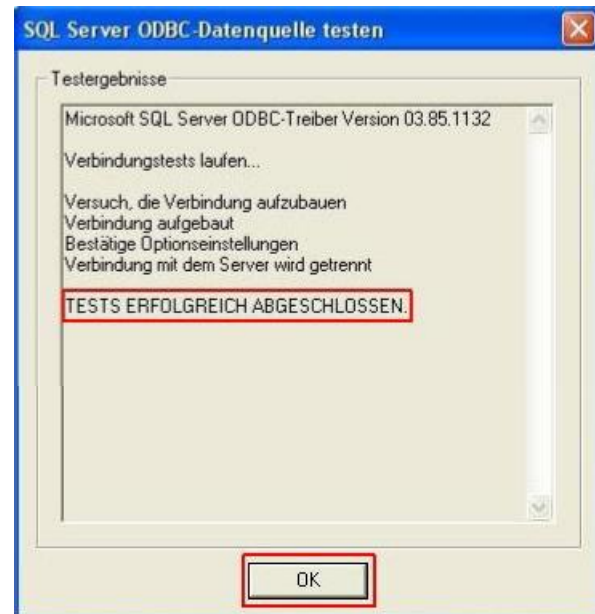


Abbildung 45: Datenquellentest OK

9 Zusammenfassung

In der Zusammenfassung wird noch einmal auf die entstanden Konflikte und ihre Änderungen sowie Verbesserungsvorschläge eingegangen.

9.1 Änderungen

Der Variablentyp text aus der Datenbank funktionierte nicht richtig. Im InTouch Projekt wurde immer nur der erste Buchstabe der Zelle angezeigt. Es stellte sich heraus, dass InTouch nur drei Datentypen unterstützt, Character(char), Integer (int) und Gleitkommazahlen (float). Es wurden die Spaltennamen vom Datentyp text durch char(10) ersetzt. Im WinCC wurde überprüft, ob die Daten auch in eine char(10)-Zelle schreiben. Es gab keine Schwierigkeiten. Die Änderung konnte beibehalten werden.

9.2 InTouch

9.2.1 Runtime

Probleme gab es beim öffnen der Datenbank, im InTouch Projekt. Es wurde ein Fenster-skript mit der Anweisung, die Datenbank zu öffnen, geschrieben (Kapitel 7.6 Seite 32). Bei InTouch kam es zu einem Laufzeitfehler.

InTouch stürzte ab und das Projekt konnte nicht mehr geöffnet werden. Es wurde ein neues Projekt angelegt und die alten Fenster importiert. Die Skripte waren noch vorhanden, mussten aber überarbeitet werden.

9.2.2 Aktionsfenster

Im Aktionsfenster vom "Zurück"-Button wird die Anweisung zum Fensterwechsel in das Aktionsskript mit eingebunden werden. Da man nur ein Aktionspunkt im Menü Schaltflächen ausgewählt werden.

9.3 externer Datenbankserver

Am Anfang wurde ein externer Datenbankserver verwendet. Da dieser technische Probleme aufwies, wurde auf die Software MS SQL Server 2005 umgestiegen. Im anschließenden Unterkapitel wird die Herstellung einer externen Datenbankverbindung kurz erklärt.

9.3.1 Datenbankbenutzer

Um eine neue Datenbank anzulegen, ist es notwendig, zuerst einen neuen Benutzer anzulegen. Über den Pfad localhost \ Security \ Logins wird mit rechtem Mausklick der Menüpunkt New Login ausgewählt.

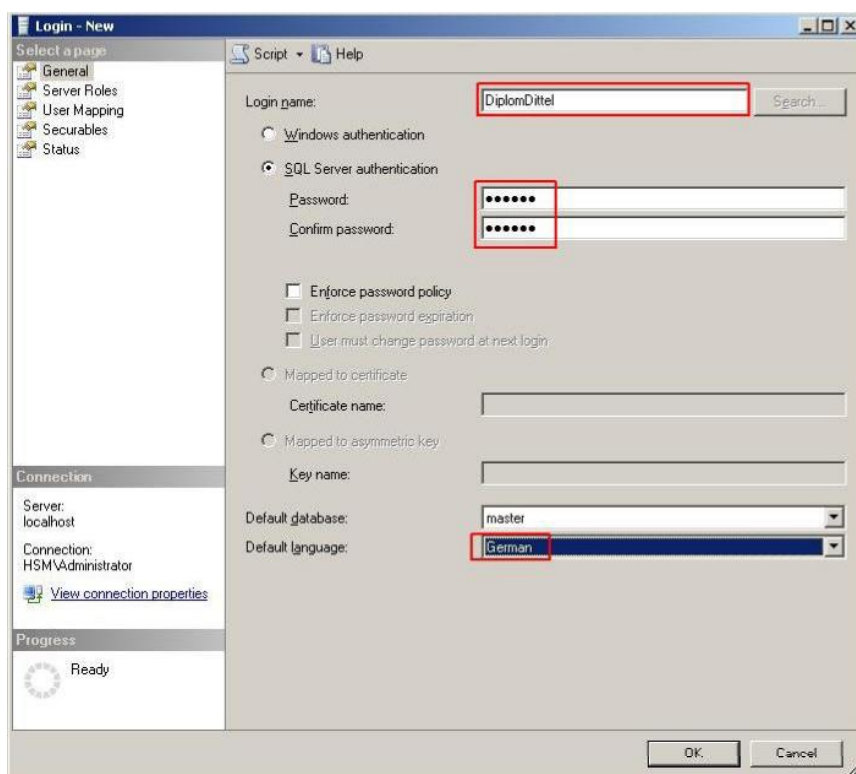


Abbildung 46: Datenbankbenutzer

Mit der Umstellung von Windows authentication zu SQL Server authentication kann der Datenbankzugriff nur mit einem Passwort im SQL-Befehl erfolgen.

Das Passwort wird für spätere Zuweisung in der Datenbankanbindung benötigt.

Die Standarddatenbank kann vorerst master bleiben, sie wird später geändert.

Durch Bestätigen mit OK wurde ein neuer Benutzer angelegt.

Nachdem der neue Benutzer angelegt ist, wird die Datenbank erstellt.

9.3.2 Datenquelle erstellen

Es wurde ebenfalls der ODBC verwendet.

Als neue Datenquelle wird ein SQL-Server erstellt. Als Servername wurde WinCCServer zugewiesen. Der Servername ist wichtig für die Verbindung zwischen WinCC und InTouch mit dem SQL-Server. In Beschreibung wird nur ein Kommentar hinzugefügt.

Die Serverbezeichnung muss per Hand eingegeben werden. Es wird die Verbindungs-IP eingetragen (Fehlerbeschreibung Kapitel 9.3.3 Seite 44).



Abbildung 47: Neue Datenquelle

Alle weiteren Einstellung sind in Abbildung 47 bis Abbildung 51 dargestellt. Zu beachten ist die SQL Server Verbindung. Einzustellen hier ist der Benutzername und das Kennwort, welches in Kapitel 9.3.1 Abbildung 46 vergeben wurde.



Abbildung 48: Datenquelle Authentizität

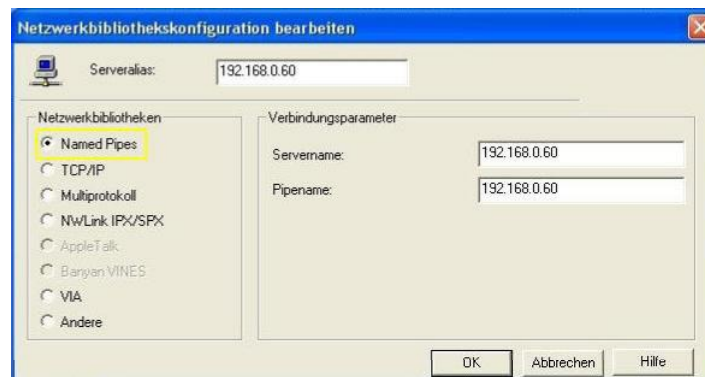


Abbildung 49: Clientkonfiguration

Die Standarddatenbank ist auf WinCCInTouch zu ändern.

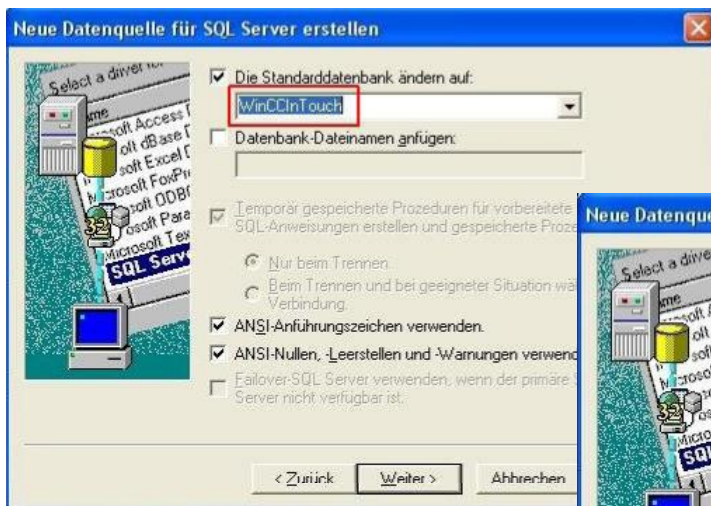


Abbildung 50: Standarddatenbank

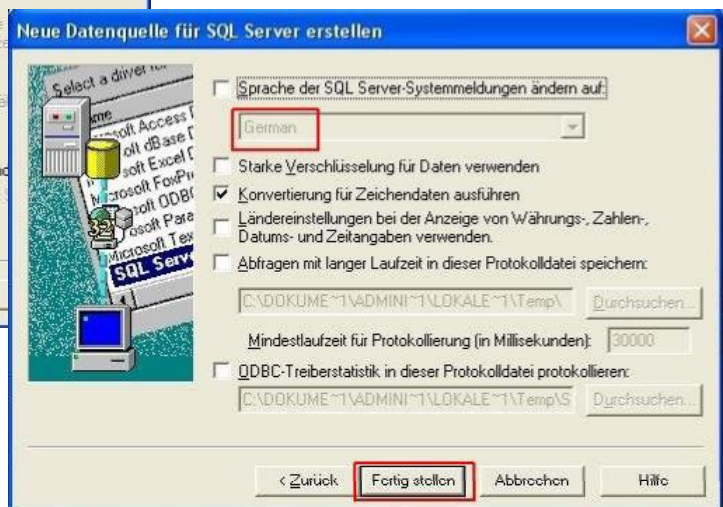


Abbildung 51: weitere Einstellungen

Nachdem die Standardsprache auf die gewünschte Sprache eingestellt wurde (hier German) kann die Konfiguration fertig gestellt werden.

9.3.3 Verbindungsfehler

Ein Verbindungsfehler (Abbildung 52) kam zustande, als der SQL-Server (Kapitel 9.3.3, Seite 44, Abbildung 52) falsch eingegeben wurde.

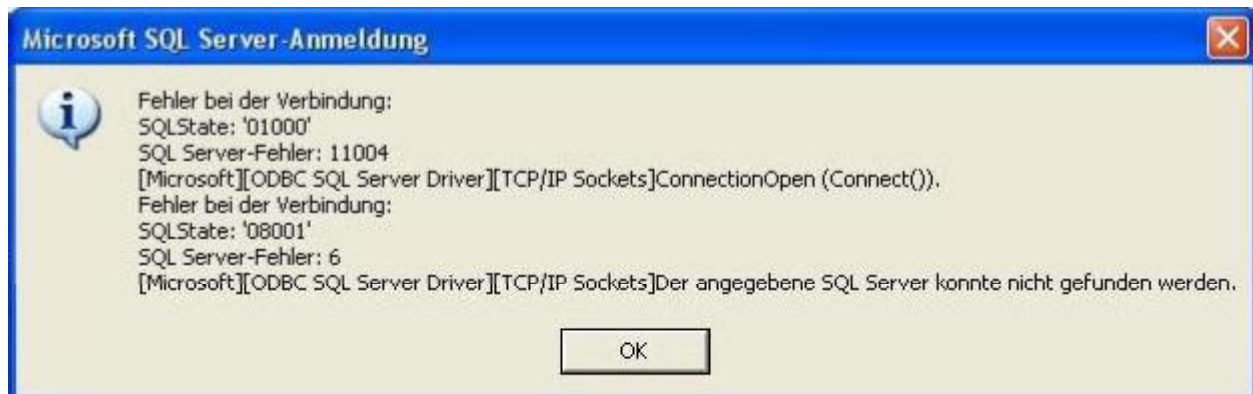


Abbildung 52: Verbindungsfehler

Nach einer langen Fehlersuche wurde festgestellt, dass die vorgegebenen Servernamen nicht korrekt aufgelöst werden konnte. Es muss, per Hand, die Server IP-Adresse angegeben werden, um die Verbindung vom Client aufzubauen.

9.4 Ausblicke

Die Hardwarekomponenten funktionieren sehr gut miteinander. Alle gewünschten Anforderungen wurden erfolgreich erzielt.

Die SPS-Werte wurden, vom WinCC aus gesteuert, in die Datenbanktabellen übernommen. Zuerst wird eine MPN aus der Datenbanktabelle MPN gelesen und um eins erhöht und in der Datenbanktabelle MPN geändert. Es wurde, bei Messstart, die MPN mit Startdatum, Startzeit, Endzeit und Enddatum in die Tabelle Zeit eingetragen. Während der Messung, werden die SPS-Werte in die messreihenabhängigen Tabellen eingetragen.

InTouch liefert die, vom Benutzer, angeforderten Werte aus der jeweiligen Datenbanktabelle. Das Projekt wurde bedienerfreundlich projektiert.

Ein Verbesserungsvorschlag wäre die Verwendung eines externen Server. Die Vorbereitungen wurden bereits begonnen (Kapitel 9.3 Seite 41).

Ein weiterer Verbesserungsvorschlag bezieht sich auf die Messwertaufnahme. Als Alternative könnte man die SPS so programmieren, dass sie in einem bestimmten Zeitraum bzw. nach Anweisung die Messungen durchführt und ein Startbit bzw. ein Endbit setzt.

Literaturverzeichnis

[1] Habermann, Matthias; Weiß Torsten: STEP 7- Crashkurs: Einführung und Vertiefung in die STEP 7-Programmiersprache. 5. Auflage Berlin, Offenbach: VDE VERLAG GMBH, 2006

Hochschulschriften:

[2] Prof. Dr.-Ing. Römer, Dietmar: Vorlesungsskript Industrielle Steuerung 1. 133 Seiten Mittweida, Hochschule Mittweida, FB Informationstechnik & Elektrotechnik

[3] Pfeffing, Theresa: Erstellung eines Rezepturverwaltungsprozesses und SQL-Datenanbindung - 2004 - 82 S.

Mittweida, Hochschule Mittweida, FB Informationstechnik & Elektrotechnik, Diplomarbeit, 2004

[4] Dittel, Nicole: WinCC-Oberfläche mit Datenbankbindung und SPS-Anbindung für Medienuntersuchung im Grenzfall -2010 - 63 S.

Mittweida, Hochschule Mittweida, Fakultät Informationstechnik & Elektrotechnik, Bachelorarbeit, 2010

Quellen aus dem Internet:

[5] Homepage Wonderware: Wonderware InTouch HMI.

URL: <<http://global.wonderware.com/DE/Pages/WonderwareInTouchHMI.aspx>>, verfügbar am 16.09.2011

Sonstiges:

[6] InTouch HMI-Hilfe aus der Software InTouch Version 10.1.300 1412.0130.0268.0005

Anhang

A AWL

AWL vom OB1(SPS 1)

```
L PEW 272  
ITD  
L 276  
/D  
T DB1.DBW 0  
DTR  
T MD 20  
L PEW 274  
ITD  
L 276  
/D  
T DB1.DBW 2  
BE
```

AWL vom OB1 (SPS 2)

```
L PEW 288  
ITD  
L 276  
/D  
T DB2.DBW 0  
DTR  
T MD 20  
L PEW 290  
ITD  
L 276  
/D  
T DB2.DBW 2  
BE
```

AWL vom OB1 (SPS 3)

```
L DB3.DBW 0  
DTR  
T MD 60  
L PEW 274  
ITD  
L 276  
/D  
T DB3.DBW 2  
BE
```


B WinCC

"Messung starten" Button (Ereignis/Maus/Mausklick)

```
Sub OnClick(Byval Item)

Dim objConnection
Dim objCommand
Dim objRecordset
Dim strConnectionString
Dim strSQL
Dim lngValue
Dim mpn_db, mpn
Dim lngCount

strConnectionString = "Provider=MSDASQL;DSN=intern;UID=local;PWD="
strSQL = "select MPN from MPN"
Set objConnection = CreateObject("ADODB.Connection")
objConnection.ConnectionString = strConnectionString
objConnection.Open
Set objRecordset = CreateObject("ADODB.Recordset")
Set objCommand = CreateObject("ADODB.Command")

objCommand.ActiveConnection = objConnection
objCommand.CommandText = strSQL
Set objRecordset = objCommand.Execute
lngCount = objRecordset.Fields.Count
If (lngCount>0) Then
    objRecordset.movefirst
    mpn_db = objRecordset.Fields(0).Value
Else
    HMIRuntime.Trace "Selection returned no fields" & vbNewLine
End If
Set objCommand = Nothing
objConnection.Close
Set objRecordset = Nothing
Set objConnection = Nothing

mpn = mpn_db +1
HMIRuntime.Tags("mpn").Write mpn

HMIRuntime.Tags("szeit").Write Time
HMIRuntime.Tags("sdatum").Write Date

HMIRuntime.Tags("starten").Write 0
HMIRuntime.Tags("beenden").Write 0

End Sub
```

"Messung beenden" Button (Ereignis/Maus/Mausklick)

```
Sub OnClick(Byval Item)

Dim objConnection
Dim objCommand
Dim strConnectionString
Dim strSQL

Dim mpn
Dim sz, sd, ez, ed

mpn = HMIRuntime.Tags("mpn").Read
sz = HMIRuntime.Tags("szeit").Read
sd = HMIRuntime.Tags("sdatum").Read
HMIRuntime.Tags("ezeit").Write Time
ez = HMIRuntime.Tags("ezeit").Read
HMIRuntime.Tags("edatum").Write Date
ed = HMIRuntime.Tags("edatum").Read

'Zeiten in Datenbank eintragen
strConnectionString = "Provider=MSDASQL;DSN=intern;UID=;PWD=;"
strSQL = "INSERT INTO Zeiten (MPN, Startzeit, Startdatum, Endzeit, End-
datum) VALUES (" & mpn & ", '" & sz & "', '" & sd & "', '" & ez & "', '"
& ed & "');"
Set objConnection = CreateObject("ADODB.Connection")
objConnection.ConnectionString = strConnectionString
objConnection.Open
Set objCommand = CreateObject("ADODB.Command")
With objCommand
.ActiveConnection = objConnection
.CommandText = strSQL
End With
objCommand.Execute
Set objCommand = Nothing
objConnection.Close
Set objConnection = Nothing

'MPN ändern
strConnectionString = "Provider=MSDASQL;DSN=intern;UID=;PWD=;"
strSQL = "UPDATE MPN SET MPN = " & mpn & ";"
Set objConnection = CreateObject("ADODB.Connection")
objConnection.ConnectionString = strConnectionString
objConnection.Open
Set objCommand = CreateObject("ADODB.Command")
With objCommand
.ActiveConnection = objConnection
.CommandText = strSQL
End With
objCommand.Execute
Set objCommand = Nothing
objConnection.Close
Set objConnection = Nothing
```

```

HMIRuntime.Tags("beenden").Write 1
HMIRuntime.Tags("starten").Write 1
End Sub

```

"Messung beenden" Button (Eigenschaften/Sonstige/Anzeige)

```

Function Visible_Trigger(Byval Item)

Dim objConnection
Dim strConnectionString
Dim strSQL
Dim objCommand

'Variablen von SPS
Dim istwert, sollwert
Dim mpn
Dim start, ende

start = HMIRuntime.Tags("starten").Read
ende = HMIRuntime.Tags("beenden").Read

If start = 0 Then
    Visible_Trigger = True
Else
    Visible_Trigger = False
End If

mpn = HMIRuntime.Tags("mpn").Read

If ende = 0 Then

'Werte von SPS holen
istwert = HMIRuntime.Tags("Istwert1").Read
sollwert = HMIRuntime.Tags("Sollwert1").Read

'Werte In Datenbank eintragen
strConnectionString = "Provider=MSDASQL;DSN=intern;UID=;PWD=;"
'Werte In Datenbank schreiben
strSQL = "INSERT INTO Fuell1 (MPN, Istwert, Sollwert) VALUES (" & mpn &
" , " & istwert & " , " & sollwert & " );"

Set objConnection = CreateObject("ADODB.Connection")
objConnection.ConnectionString = strConnectionString
objConnection.Open
Set objCommand = CreateObject("ADODB.Command")
With objCommand
.ActiveConnection = objConnection
.CommandText = strSQL
End With
objCommand.Execute
Set objCommand = Nothing

```

```
objConnection.Close
Set objConnection = Nothing
Else
End If

End Function
```

C InTouch-Skripte

Startbild: *Button*

```
RESULTCODE = SQLConnect (IDNr, "DSN=intern");
Show "Suchfenster";
```

Suchfensterbild: *Fensterskipt*

```
IF Anzahl > 0 THEN
    anzeigen = 1;
ELSE
    anzeigen =0;
ENDIF;
```

"Nach Datensätzen suchen" Button

```
SQLSelect (IDNr, "Zeiten","Zeiten","Startdatum= '"+ Datum + "'", "");
Anzahl = SQLNumRows (IDNr);
```

"Prev" Button

```
SQLPrev (IDNr);
```

"Next" Button

```
SQLNext (IDNr);
```

"gefundene Werte anzeigen" Button

```
SQLFirst (IDNr);
```

"Anlage anzeigen" Button

```
SQLSelect (IDNr, "Fuell1", "Fuell1", "MPN="+Text (MPN_int, "#.##"), "");
f1 = SQLNumRows ( IDNr);
SQLSelect (IDNr, "Fuell2", "Fuell1", "MPN="+Text (MPN_int, "#.##"), "");
f2 = SQLNumRows ( IDNr);
SQLSelect (IDNr,
"Temperatur", "Temperatur", "MPN="+Text (MPN_int, "#.##"), "");
f3 = SQLNumRows ( IDNr);
IF f1>0 THEN
    anlage = "Füllregelung1";
    anzeigen2 = 1;
ELSE
    IF f2>0 THEN
        anlage = "Füllregelung2";
        anzeigen2 = 1;
    ELSE
        IF f3>0 THEN
            anlage = "Temperaturregelung";
            anzeigen2 = 1;
        ELSE
            anlage = "MPN überprüfen";
            anzeigen2 = 0;
        ENDIF;
    ENDIF;
ENDIF;
```

"anzeigen" Button

```
IF f1>0 THEN
    Show "Fuellanlage1";
ELSE
    IF f2>0 THEN
        Show "Fuellanlage2";
    ELSE
        IF f3>0 THEN
            Show "Temperaturregelung";
        ELSE
            anlage = "FEHLER";
        ENDIF;
    ENDIF;
ENDIF;
```

"Drucken" Button

```
PrintScreen(1,4);
```

"Zurück" Button

```
anzeigen=0;
anzeigen2=0;
anlage="";
Anzahl=0;
Datum="";
f1=0;
f2=0;
f3=0;
MPN_int=0;
MPN=0;
Uhrzeit="";
Uhrzeit_DB="";
RESULTCODE = SQLDisconnect (IDNr);
Show "Start";
```

Fuellanlage1: *"erster Wert" Button*

```
SQLSelect (IDNr, "Fuel11", "Fuel11", "MPN="+Text (MPN, "#.##"), "");
[SQLSelect (IDNr, "Fuel12", "Fuel11", "MPN="+Text (MPN, "#.##"), "");]
[SQLSelect (IDNr,
    "Temperatur", "Temperatur", "MPN="+Text (MPN, "#.##"), "");]
SQLFirst (IDNr);
```

D Quelltexterläuterung

WinCC Quelltext

Programmtext	Funktion
Dim	innerhalb eines Skripts zur Variablendefinition
Function ... End Function	Funktionsaufruf
HMIRuntime.Tags("Variable").Read	Wert aus einer <i>Variable</i> auslesen
HMIRuntime.Tags("VariableA").Write <i>VariableB</i>	Schreibe Wert aus <i>VariableB</i> in <i>VariableA</i>
HMIRuntime.Tags("Variable").Write <i>x</i>	Schreibe Wert <i>x</i> (<i>definierter Wert</i>) in <i>Variable</i>
If <i>Variable1</i> = <i>Variable2</i> Then ... Else ... End If	Wenn <i>Variable1</i> = <i>Variable2</i> Dann ... Sonst ...
obj.Connection.Close	Datenbank schliessen
obj.Connection.Open	Datenbank öffnen
Set objConnection ... Set objConnection	Datenbankanweisungen
strConnectionString="Provider=MSDAS- QL;DSN= <i>Datenbankname</i> ;UID=local;PW D=;"	Datenbank <i>Datenbankname</i> aufrufen
strSQL = "INSERT INTO "& <i>TName</i> &" (<i>MPN</i>) VALUES ("& <i>mpn</i> &");"	Variablenwert <i>mpn</i> in Spalte <i>MPN</i> in die Da- tenbanktabelle mit Wert von <i>TName</i> eintra- gen
strSQL = "select <i>Variable</i> from <i>Tabellen-</i> <i>name</i> "	Datenbankanweisung; holt <i>Variablenwert</i> aus <i>Tabelle</i>
strSQL = "UPDATE MN SET MPN " & <i>mpn</i> &" "	MPNwert in der Datenbank ändern (über- schreiben)
Sub OnClick(Byval Item)	Button-Drück-Funktion
Sub ... End Sub	Funktionsaufruf
<i>Variable</i> = <i>x</i>	<i>Variable</i> bekommt den Wert <i>x</i>

InTouch SQL Syntax [6]

SQLConnect(VerbindungsID, "Verbindungszeichenfolge" ¹);
SQLSelect(VerbindungsID, Tabellename, Bindeliste, Wenn Ausdruck, Ordnen);
Show Fenstername
SQLNumRows(VerbindungsID);
SQLFirst(VerbindungsID);
SQLPrev(VerbindungsID);
SQLNext(VerbindungsID);
SQLDisconnect(VerbindungsID);
PrintScreen(Bildschirmoption ² , Druckoption ³);

InTouch Quelltextbedeutung

RESULTCODE=SQLConnect(IDNr, "DSN=intern");	Datenbank öffnen
Show "Suchfenster"	Fenster "Suchfenster" öffnen
IF Anzahl >0 Then anzeigen = 1; ELSE anzeigen = 0; ENDIF	Wenn Anzahl >0 dann Variable anzeigen =1 sonst anzeigen =0 Schleifenende
SQLSelect(IDNr,"Zeiten","Zeiten",Startdatum="Datum+""; "");	Datensatz aus der Datenbanktabelle Zeiten, wenn Bedingung Startdatumspalte = Datumvariable ist, Werte holen und in Bindeliste Zeiten einfügen.
Anzahl=SQLNumRows(IDNr);	Anzahl der Datenbank-Tabellen-Zeilen wird in Variable Anzahl geschrieben
SQLFirst(IDNr);	erster Datensatz
SQLPrev(IDNr);	vorheriger Datensatz
SQLNext(IDNr);	nächster Datensatz
RESULTCODE=SQLDisconnect(IDNr);	offene Datenbank schließen
PrintScreen(1,4);	drucken des Fensters

1 DSN für Datenbankmanager und DB für Datenbank

2 1 - Client-Bereich, keine Menüs; 2 - gesamter Fensterbereich

3 1 - bestmöglich, 2 - vertikal anpassen, 3 - horizontal anpassen, 4 - gesamte Seite füllen